

L'INDISPENSABLE

POUR



ATARI ST

A. RAVET





Collection marabout service



Afin de vous informer de toutes ses publications, marabout édite des catalogues où sont annoncés, régulièrement, les nombreux ouvrages qui vous intéressent. Vous pouvez les obtenir gracieusement auprès de votre libraire habituel.

**Atari est une marque déposée.
Tous les noms cités dans cet ouvrage sont des marques
déposées.**

© 1988 by Marabout, Allier (Belgique)

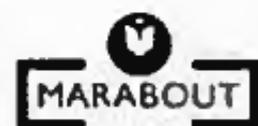
**Toute reproduction d'un extrait quelconque de ce livre par quelque procédé que ce
soit, et notamment par photocopie ou microfilm est interdite sans autorisation écrite
de l'éditeur.**

Alain Ravet

L'Indispensable

pour

Atari ST





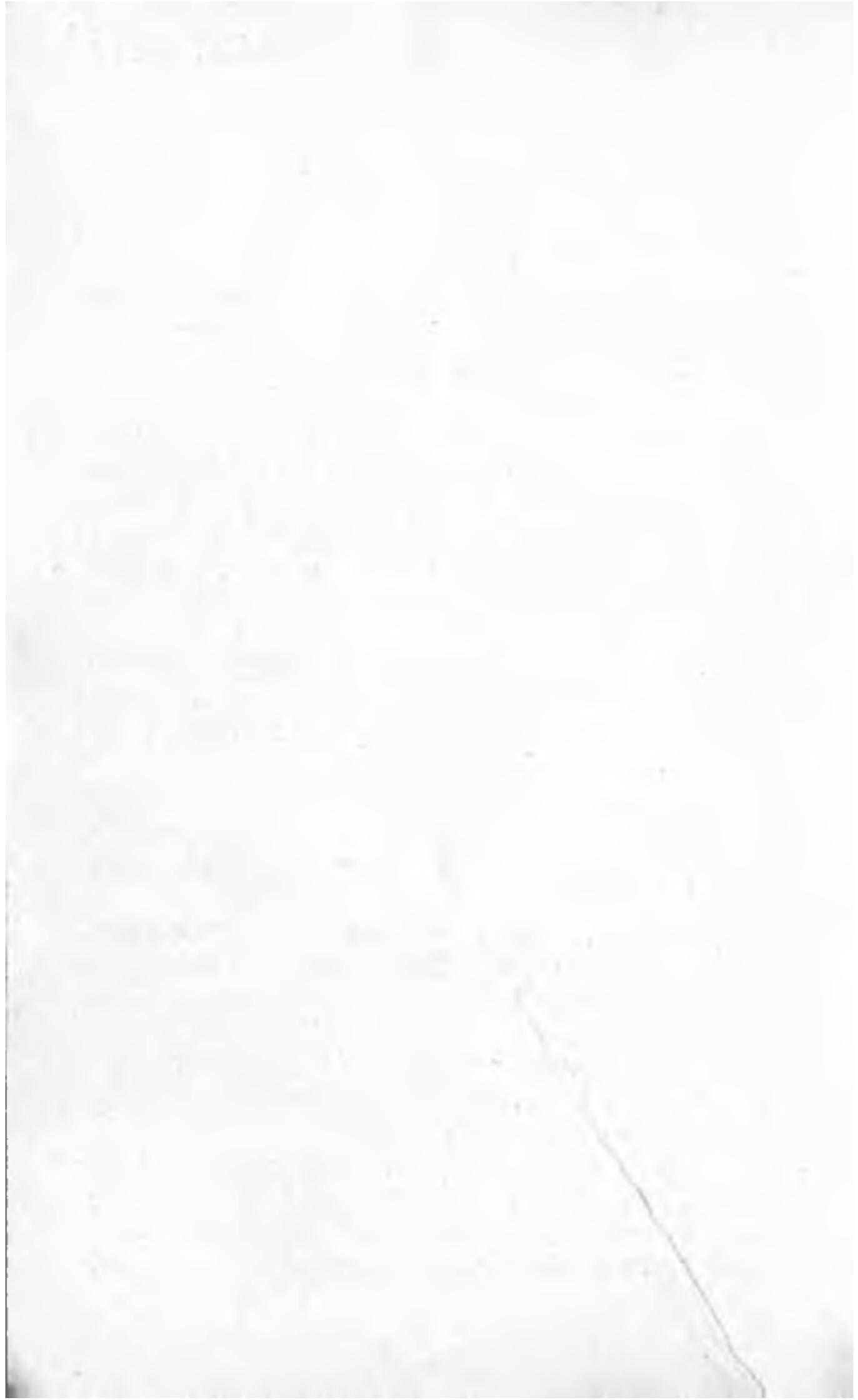
Sommaire

Introduction	9
Généralités	11
Le système d'exploitation	45
Gem	61
Les langages de programmation	115
Le graphisme	155
Les traitements de textes	177
La musique et la norme MIDI	195
Les périphériques	201
Les utilitaires	247
Les virus	255
Le hardware	263
Annexes	273
Table des matières	279



INTRODUCTION

112-701



Que vous possédiez ou non un ATARI ST, que vous soyez néophyte ou chevronné, converti ou hésitant, ce livre est écrit pour vous.

Vous y apprendrez peut-être ce que vous avez toujours voulu savoir sans jamais oser le demander ou y découvrirez le nom et les coordonnées du programme que vous cherchez désespérément depuis des mois. Le débutant trouvera dans l'annexe un rappel des notions de bases de l'informatique.

L'ATARI ST est présenté et analysé dans le chapitre *GÉNÉRALITÉS*. Son histoire, ses possibilités, ses limites et ses points forts y sont décrits en détail.

Les chapitres *SYSTEME D'EXPLOITATION* et *GEM* traitent de caractéristiques logicielles du ST, tandis que le chapitre *HARDWARE* analyse ses spécificités matérielles.

Dans le chapitre *LANGAGES DE PROGRAMMATION*, vous trouverez la liste des langages disponibles sur ATARI ST, leur histoire, leurs caractéristiques et leurs limites.

Les chapitres *TRAITEMENT DE TEXTE*, *MUSIQUE* et *GRAPHISME* introduisent et traitent en détail ces trois types d'application.

Comme son nom l'indique, le chapitre *PÉRIPHÉRIQUES* traite et explique les périphériques de toutes sortes. Après l'avoir lu, imprimante laser, disque optique, digitaliseur sonore, scanner, moniteur

géant et lunettes stéréoscopiques n'auront plus de secret pour vous. Vous saurez ce qu'ils font et comment ils le font.

Dans *UTILITAIRES* sont décrits et expliqués tous ces petits programmes qui facilitent la vie de l'utilisateur ainsi que ceux qui transforment votre ATARI ST en *compatible* IBM-PC ou *compatible* MACINTOSH.

Et pour terminer, vous apprendrez dans *VIRUS* tout de cet incidieux "micro-organisme", de ses méthodes d'actions perverses ainsi que des soins post-infectieux et des vaccins qui nous en préservent.

Que vous soyez débutant ou chevronné, ce livre est écrit pour vous. En espérant que vous prendrez plaisir à le lire et qu'il vous apprendra un tas de choses intéressantes, je vous souhaite une bonne lecture.

GÉNÉRALITÉS



LA GAMME ST :

Un peu d'histoire :

1984 . L'année de Big Brother. Qui aurait pu se douter alors, un an avant la sortie du premier ATARI ST, que ce fabricant de jeux vidéo sur le déclin sortirait sous peu de ses cartons une machine révolutionnaire, aux prétentions professionnelles, capable un jour de faire la nique aux plus grands ? Personne n'aurait osé miser le moindre sou sur cette firme au bord de la faillite. Les employés pleuraient, les créanciers tremblaient, les voisins s'inquiétaient, et alors ... et alors ... Zorro est arrivé ! Eh oui, comme je vous le dis !

Enfin, pas tout à fait. Ce n'était pas le vrai Zorro, mais c'est tout comme : c'était Jack Tramiel. Il était à l'époque le président fondateur de COMMODORE - un des précurseurs de la micro-informatique avec son PET-CBM - et n'a pas hésité à vendre sa firme en plein essor, pour retaper ce mouton à trois pattes auquel personne ne croyait plus.

Et il a gagné son pari.

☛ Une idée :

L'idée de base qui a guidé la création du ST fut aussi simple qu'efficace :

Utiliser un **microprocesseur performant et moderne.**

Lui associer une kyrielle de **processeurs périphériques** (esclaves en silicium) pour le remplacer dans les tâches secondaires.

Le doter d'un **système d'exploitation pré-existant** et proche du MS-DOS.

Faciliter l'usage de la machine en plaçant entre elle et l'utilisateur une **interface graphique puissante et confortable.**

☛ Un peu de politique :

Aucun de ces choix n'a été fait au hasard. Ils correspondent tous en réalité à une politique agressive - pour les concurrents - et séduisante - pour les utilisateurs - : produire une machine qui possède toutes les qualités suivantes :

- ✓ performante;
- ✓ complète;
- ✓ confortable;
- ✓ uniforme;
- ✓ ouverte sur le monde MS-DOS et
- ✓ peu coûteuse.

Le résultat : l'ATARI ST.

☛ PERFORMANCE

Le microprocesseur central (la puce) qui fait battre le coeur de l'ATARI ST est le **MOTOROLA MC68000**. Il a été préféré à son concurrent immédiat, l'**INTEL 80xx** qui équipe les compatibles IBM, principalement parce qu'il était plus rapide, mieux conçu et d'une technologie plus récente.

L'usage systématique d'un maximum de circuits périphériques spécialisés pour toutes les activités non spécifiques au microprocesseur central lui permet de ne pas perdre de temps dans ces fonctions annexes. Ils se chargent de gérer l'écran, le clavier, la souris, la mémoire, le lecteur de disquettes et bien d'autres choses encore. C'est pour cette raison qu'un même programme, écrit pour l'ordinateur Macintosh tournera plus vite sur un Atari ST muni d'un émulateur que sur un Macintosh. Il est pourtant lui aussi équipé du même MC68000, mais ne dispose pas d'autant d'esclaves (voir le chapitre consacré aux *UTILISATEURS: Émulateurs*).

☛ COMPLÉTUDE

Il fut une époque - pas tout à fait révolue - où l'achat d'un ordinateur nécessitait une véritable liste d'épiciers : d'abord, on demandait au vendeur une "unité centrale". Et puis on commandait un clavier. Après on choisissait un écran et son indispensable carte d'interfaçage. Si par la suite on désirait une imprimante, il fallait y ajouter une carte d'interfaçage séparée. Pour le second lecteur de disquettes, idem,

ainsi que pour le disque dur et le modem. Si l'on souhaitait ultérieurement employer un écran couleur à la place du monochrome, il fallait, en plus du moniteur, remplacer la carte graphique, en espérant que tous les programmes achetés entre-temps l'accepteraient sans rechigner.

Tel n'est pas le lot du possesseur d'un ATARI ST.

Tout modèle de la gamme ST, au sortir de sa boîte, est *prêt à l'emploi*: il possède au moins un lecteur de disquettes, une souris, un clavier et un moniteur, couleur ou monochrome, et plusieurs logiciels l'accompagnent, tels qu'un interpréteur basic, un traitement de texte et quelques autres encore. (*Remarque* : certains modèles pouvant être raccordés à un téléviseur courant, une version "économico-ludique" existe et est vendue sans moniteur. C'est la configuration idéale pour ceux que seuls les jeux intéressent.)

Un coup d'oeil au dos d'une des machines de la gamme nous révèle qu'elles sont directement connectables à tous les périphériques habituels, et même plus.

On dispose donc à l'origine des circuits internes pour piloter et des connecteurs externes pour se raccorder à un moniteur (écran couleur ou monochrome, au choix), un lecteur de disquettes, de un à sept disques durs, une souris et un ou deux joysticks, une imprimante classique (série ou parallèle), une imprimante laser, une table traçante, un modem, une tablette graphique (voir le chapitre consacré aux *PERIPHERIQUES*).

Les concepteurs ont ajouté à cela 2 connecteurs - **MIDI-IN** et **MIDI-OUT** - et une puce spécialisée per-

mettant au ST de commander directement un ou plusieurs synthétiseurs (voir le chapitre consacré à la *MUSIQUE*).

Le port cartouche est situé sur le côté gauche de l'ordinateur. Il donne accès à toute une gamme de périphériques spéciaux (lunettes stéréoscopiques, horloge permanente, ...) ou à des logiciels disponibles directement sur cartouche. C'est là aussi que se place le *dongle*, petite boîte contenant, scellé dans une résine opaque, un circuit électronique qui cache une clé. Elle est unique, secrète, incopiable et nécessaire au fonctionnement de certains logiciels. C'est la protection anti-piratage la plus efficace et la plus confortable pour l'utilisateur.

Regardons ces connecteurs de plus près :

Les modèles 520 STF, 1040 STF et MEGA ST possèdent un transformateur intégré au boîtier principal. Ils peuvent donc être reliés directement au secteur.



Les modèles 260 ST et 520 ST sont alimentés par un transformateur externe.



18 / Généralités

On peut connecter indifféremment un moniteur couleur ou monochrome; le système reconnaît automatiquement son type.



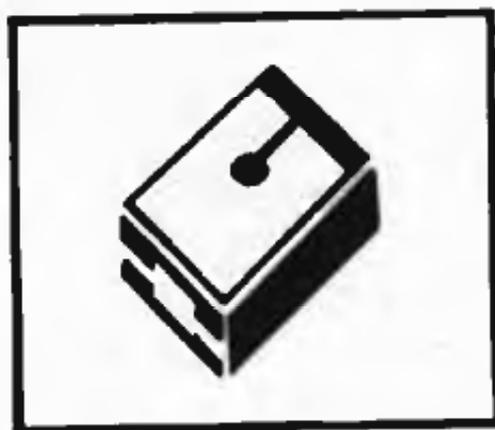
Remarque : si l'on dispose d'un téléviseur avec une prise péritel, il peut être connecté ici directement à tous les modèles de la gamme au moyen d'un cordon ad hoc (ATARI-> Péritel), non fourni à l'origine, mais disponible sans problème.

Un ATARI ST, quel qu'il soit, peut être équipé au maximum de 2 lecteurs de disquettes.

Les modèles 1040 STF et MEGA ST disposent à l'origine d'un premier lecteur de disquettes (drive) intégré à l'unité centrale.

On peut en ajouter un second, externe, qui se branchera dans ce connecteur.

Les modèles 520 ST ne disposent pas de lecteur interne. Il faut raccorder le premier lecteur à ce connecteur. Si l'on veut une configuration à 2 lecteurs de disquettes il faudra les chaîner.



C'est la sortie **Centronics** standard qui permet de raccorder toutes les imprimantes du type EPSON ou IBM compatible.



C'est ici également que peuvent se raccorder certaines **cartes d'extension** reliant l'ordinateur au monde extérieur.

C'est la sortie **RS-232** standard permettant non seulement de connecter un modem, mais aussi une imprimante série ou tout autre périphérique ayant besoin d'une liaison série (voir les *ANNEXES*).



Ce connecteur est destiné à la **souris**, fournie à l'origine avec le système, mais peut aussi accepter un **joystick**.



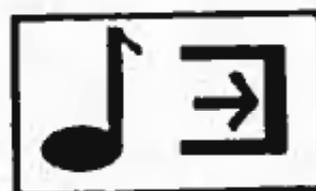
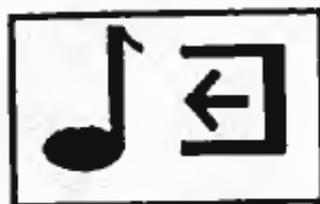
Ce connecteur accepte uniquement la connexion d'un **joystick**. Il est connu du système comme n°1, tandis que celui branché sur le connecteur souris est le n°2.



Ce connecteur sert non seulement au **disque dur**, mais aussi à l'**imprimante laser** ou au **lecteur CD-ROM**. Ces périphériques peuvent aussi être connectés ensemble par chaînage, comme les lecteurs de disquettes.



Ces connecteurs permettent la connexion à tous les appareils **MIDI** (voir le chapitre consacré à la **MUSIQUE**), mais ils ont aussi été détournés pour d'autres applications, non musicales, comme, par exemple, un réseau économique.



☛ CONFORT

La première marque de confort apparaît directement lors du déballage de l'ordinateur et de son installation : tout étant installé à l'origine, il n'y a pas de capot à soulever ni de carte à brancher. Le manuel d'installation est presque superflu et le perdre n'est pas un problème.

Chaque connecteur est accompagné d'un logo significatif, indiquant son usage précis; pas besoin donc, le jour où l'on veut connecter un modem de retrouver la notice rangée deux ans plus tôt. Étant totalement incompatibles entre eux, il est impossible de commettre une irréparable gaffe, comme connecter le 220 Volts sur l'entrée de la souris ou de faire imploser l'écran.

L'aspect le plus confortable des ST est peut-être le fait que le système d'exploitation (O.S. : Operating System) est contenu à l'origine dans la mémoire de l'ordinateur (en ROM), et ne doit donc pas être préalablement chargé à partir d'une disquette. Il est disponible presque instantanément, dès la mise sous tension de la machine (voir le chapitre consacré à l'OS).

L'interface graphique - le GEM - est elle aussi en ROM et toujours disponible.

Basée sur un système de *fenêtres, icônes, menus* et *pointeurs*, elle rend instinctif l'usage du système et permet à tous de se servir de l'ordinateur sans presque aucune connaissance prérequis. Si ces termes vous intriguent, ne vous en faites pas : ils

seront expliqués en long et en large dans le chapitre consacré au GEM.

La mémoire spacieuse, de 520 Ko à 4 mégaoctets, suivant le modèle, est d'autant plus agréable qu'elle est presque entièrement réservée à l'utilisateur. En effet, le système d'exploitation - l'OS - et l'interface graphique - le GEM - se trouvant en ROM, ils ne doivent donc pas être recopiés en RAM comme sur la plupart des ordinateurs.

☞ Remarque

En informatique, il y a deux concepts qui sont mutuellement exclusifs: l'espace et le temps, autrement dit, la taille du programme plus la mémoire qu'il emploie pour travailler et la rapidité à laquelle il tourne. Pour optimiser ses performances, dans un langage donné et en supposant qu'il était déjà bien conçu, il faut le réécrire en lui faisant employer plus d'espace pour ses calculs intermédiaires.

☞ UNIFORMITÉ

L'uniformité, ou homogénéité matérielle, entre les différents modèles de la gamme est garantie quelle que soit la configuration adoptée (taille mémoire, type de moniteur, nombre de lecteurs), par l'aspect monolithique et complet des machines de la série ST. Examinons d'abord tous les modèles de la gamme avant de constater que les différences qui les distinguent ne

sont pas un obstacle pour l'utilisateur ou le concepteur de logiciels.

Dans notre analyse, nous suivrons le schéma suivant :

1. Les points communs.
2. Les différents modèles.
3. Les différences et leurs conséquences.
4. L'uniformité et ses avantages.

1. Les points communs

Comme nous l'avons vu précédemment, tous les modèles sont construits autour des éléments suivants:

- ✓ un MC68000 puissamment secondé;
- ✓ une souris;
- ✓ une série complète de connecteurs;
- ✓ un clavier standard (variant suivant les pays);
- ✓ un OS couplé au GEM (traduit suivant les pays);
- ✓ un lecteur de disquettes 3"1/2;
- ✓ un moniteur.

o Remarque

Tout ceci se retrouve dans tous les modèles, sauf dans la version économique du 520.

Cet ensemble de base homogène a été décliné en plusieurs modèles distincts. Ceci pour plusieurs raisons :

○ **Economiques:** évolution des prix et des disponibilités des circuits de mémoires (RAM).

- **Technologiques:** mise au point de nouveaux circuits et amélioration des performances de la machine.
- **De marketing:** couverture d'un public le plus large possible, de l'amateur averti au professionnel, sans oublier le "kid" fauché mais passionné du joystick que seuls les jeux vidéo intéressent.

2. Les différents modèles

Avant de les passer en revue, expliquons l'origine des termes ST, STF, STFM et MEGA :

□ **ST :** Sixteen/Thirty two (16/32 en anglais).

Le MC68000 est un processeur 16/32 bits (voir le chapitre consacré au *HARDWARE*)

□ **STF :** ST + Floppy interne.

(Floppy drive = lecteur de disquettes)

Le premier lecteur de disquette et l'alimentation de l'ordinateur (le transformateur) sont contenus dans le boîtier de l'unité centrale, avec le clavier.

□ **STFM :** STF + modulateur TV interne.

Le modulateur TV permet d'envoyer l'image - uniquement couleur - vers un téléviseur quelconque.

□ **MEGA :**

La taille de la mémoire est un multiple du méga-octet.

Ainsi, du 260 ST au 1040 ST, tous reprennent le principe de l'ordinateur-clavier : l'unité centrale se compose d'un boîtier compact contenant la carte mère - les circuits électroniques -, les connecteurs et le clavier.

Ceci étant dit, nous pouvons énumérer les différents modèles de la gamme.

130 ST

Mort-né

Nous ne retiendrons de cette bizarrerie que son nom et sa taille mémoire "lilliputienne" : 128 Ko. La seule justification de sa programmation pour la série ST était de proposer un modèle très économique à une époque où le prix des circuits de mémoire (RAM) était encore élevé. La baisse subite et importante de ceux-ci à balayé ce projet qui n'a donc jamais vu le jour.

260 ST

Éphémère

Ce modèle économique a fait une brève apparition sur le marché. Sa petite mémoire 256 Ko apportait trop de désavantages pour qu'il résiste à la chute des prix. Il possédait l'OS en ROM (heureusement). La plupart de ses possesseurs ont étendu sa taille mémoire à peu de frais.

520 ST

Le pionnier

Véritable point d'entrée dans la famille, c'est sur le 520 ST que les concepteurs ont développé les premiers logiciels sérieux. Lors de sa sortie, en 1985, le système d'exploitation, non définitif, n'était pas encore placé dans les ROM de la machine. Il fallait donc

le charger en mémoire avant toute session de travail, ce qui prenait de la place - 190 Ko - et du temps.

Par la suite est apparu le 520 STFM, intégrant alimentation et lecteur au boîtier. Notons que, contrairement au 1040, la majorité des 520 mis sur le marché sont équipés d'un modulateur interne.

⇒ **Caractéristiques :**

- Mémoire : 512 Ko.
- Lecteur : SF, 360 Ko.
SF = simple face (SS = Single Side en anglais)
capacité = 360 k

Le lecteur est externe sur le ST, interne sur le STFM.

- Affichage : modulateur TV intégré sur (presque) tous les modèles
- Alimentation : externe sur le ST, interne sur le STFM.

☞ **Remarques:**

✓ La mémoire : il est possible de l'étendre jusqu'à 1, 2 ou même 4 Mo (méga-octets). Le 1040 le permet aussi, mais c'est sur le 520 que cette modification se fait le plus facilement.

✓ Le lecteur : depuis le milieu de 1988, tous les exemplaires de 520 STFM sont fournis munis d'un lecteur double face, comme le 1040 STF. Il est de plus tout à fait permis d'employer un lecteur externe double face avec un 520 ST.

✓ La souris : sur les ST, la souris se connectait sur le côté droit de la machine, ce qui en rendait l'usage malaisé pour les gauchers. Les STF ont corrigé ce défaut en

plaçant ce connecteur dans une position plus centrale, sous la machine.

✓ L'alimentation interne : sur les STF, le transformateur se trouve à l'intérieur du boîtier. Il n'y a pas de ventilateur, car il s'avère qu'il ne chauffe pas exagérément, même lors d'un usage prolongé de la machine. Cela reste vrai tant que l'on n'empêche pas son refroidissement qui s'effectue naturellement par un courant d'air. Il entre par les fentes situées sous l'appareil et en sort par les ouvertures de la face supérieure, camouflées dans la décoration - les stries obliques - située devant le clavier. Pour ne pas les obstruer, il est important de ne pas placer l'ordinateur sur une surface molle (feutre, tapis) et de ne rien déposer (feuilles, livre) sur son dessus.

1040 STF

⇒ Caractéristiques:

Il est identique au 520, sauf en ce qui concerne :

✓ Le lecteur : 3"1/2 (DD 720 k interne).

DD= lecteur double face : on écrit sur les 2 côtés de la disquettes.

Le lecteur interne (intégré au boîtier) est connu de l'OS comme étant le lecteur "A". Le connecteur de lecteur externe peut accepter un second lecteur (lecteur "B"), mais il n'est pas permis d'en chaîner un second - ce qui ferait 3 en tout - comme sur le 520 ST.

✓ **L' affichage :**

La majorité des 1040 sont des STF, sans modulateur interne, mais une version STFM est apparue sur le marché.

Si vous possédez un 1040 STF et que l'absence de modulateur vous empêche de dormir, sachez que vous pourrez toujours en ajouter un externe. En vente chez tous les bons distributeurs.

MEGA ST 2

MEGA ST 4

Il s'agit presque d'une autre machine. Sa carte mère redessinée, sa mémoire plus importante, son aspect plus "sérieux" et son prix plus élevé la destinent à un public plus professionnel, principalement axé sur le *graphisme* (gros consommateur de mémoire et de calculs) ou le *desktop publishing* (il faut au moins un MEGA ST 2 pour piloter l'imprimante laser d'ATARI). Certaines améliorations techniques en font une machine plus ouverte et plus performante.

De plus, l'aspect extérieur a complètement changé :

✓ **L'unité centrale :**

Elle contient la carte mère, l'alimentation et un lecteur de disquettes DF 760 Ko. Elle se place sous le moniteur.

✓ **Le clavier :**

Il est séparé de l'unité centrale, et donc plus léger et plus mobile. Il est relié à l'unité centrale par un cordon de type téléphone. Sa frappe est plus franche et plus nette car il s'agit d'un clavier mécanique.

✓ **La mémoire :**

Elle a augmenté : mémoire vive (RAM): 2 ou 4 Mo (on ne peut cependant l'étendre au-delà de 4 Mo).

✓ **Les améliorations techniques :**

L'horloge permanente: tous les modèles ST possèdent une horloge interne qui retient l'heure et la date. Dans les modèles 260, 520 et 1040, elles doivent être initialisées au début de chaque session au moyen du "CONTROL PANEL" (voir le chapitre consacré à GEM). Une fois le courant coupé, la mémoire est perdue. Les modèles MEGA ST possèdent une batterie qui préserve l'heure et la date. Ces deux paramètres doivent être initialisés une fois pour toutes lors de la première utilisation de la machine. Cette information est importante car l'OS associe à chaque fichier qu'il manipule l'heure et la date de sa dernière modification. (voir le chapitre consacré à GEM)

Le coprocesseur graphique : le "Blitter"

Il accélère l'affichage. Son rôle n'est pas négligeable car la partie de l'OS qui gère l'affichage n'est pas des plus performantes du point de vue de la rapidité de certaines opérations.

L'accès au bus : Les modèles 260, 520 et 1040 sont des machines "fermées" : il est impossible d'étendre leurs performances. Les modèles MEGA ST sont, eux, plus "ouverts" : l'accès au bus permet, entre autres, de les relier à d'autres ordinateurs (*émulateur PC* ou *ABAQ*) ou de remplacer le moniteur monochrome standard par un écran géant.

3. Différences et conséquences pour l'utilisateur :

Comme nous venons de le voir, les différents modèles de la gamme ST, bien que construits autour d'une même base, se distinguent les uns des autres par des modifications plus ou moins importantes. Voyons les conséquences qu'elles peuvent avoir pour l'utilisateur lors de l'usage de sa machine.

Les points passés en revue sont :

- ✓ la taille mémoire;
- ✓ le type de lecteur de disquettes;
- ✓ le nombre de lecteurs de disquettes;
- ✓ le type d'écran;
- ✓ les versions nationales de l'OS;
- ✓ les versions successives de l'OS;
- ✓ l'accès au bus.

3.1. Les différences de taille mémoire

On peut considérer, en première approximation, que tout logiciel tourne sur toute configuration de la mémoire. Mais comme toute règle, elle a ses exceptions. On peut donc diviser les logiciels en quatre catégories, selon leurs ajustement dans 520 Ko.

A savoir : ceux pour qui c'est :

- a. assez
- b. juste
- c. trop juste
- d. pas assez

a. Assez :

Cette catégorie regroupe tous les programmes qui fonctionnent correctement et d'une manière optimale dans la mémoire d'un 520 ST.

On y trouve, en vrac, les traitements de texte, les utilitaires, les programmes simples de dessin, les programmes de communication, les interpréteurs, les jeux, les éditeurs et bien d'autres encore.

b. Juste :

Cette seconde catégorie contient tous les programmes qui peuvent tourner correctement dans un 520 ST, mais fonctionne plus "confortablement" dans une mémoire plus spacieuse. Cette notion de confort d'utilisation est un mélange de deux critères :

- la taille des documents que l'on peut manipuler en mémoire;
- la fréquence des chargements de fichiers : si un programme est trop volumineux pour une configuration mémoire donnée, il faudra le découper en plusieurs segments - appelés OVERLAYS - et les charger au fur et à mesure des besoins.

Appartiennent à cette catégorie principalement certains logiciels de P.A.O. (Publication Assistée par Ordinateur), des logiciels graphiques et les compilateurs.

c. Trop juste :

Les programmes concernés peuvent fonctionner sur une configuration minimale - 520 Ko - mais au prix d'une perte de performances excessive : soit ils "rament" (tourment très très lentement), soit l'espace disponible pour les données à manipuler est trop petit pour en contenir une quantité suffisante. Un programme de dessin en trois dimension ne pourra, par exemple, représenter que quelques objets ou sera trop lent dans son fonctionnement.

Remarquons cependant que la limite avec la deuxième catégorie n'est pas toujours nette, et dépend parfois du type d'usage de ces programmes ou de l'ampleur des projets.

On peut citer sous toutes réserves (voir la remarque précédente) certains programmes sophistiqués d'animation de dessins en trois dimensions (3 D), ou encore un interpréteur Lisp (voir le chapitre consacré aux *LANGAGES* : Lisp).

d. Pas assez :

Il s'agit clairement de programmes qui exigent 1, 2 ou même 4 Mo (méga-octets) de mémoire vive.

L'exemple le plus net est le programme qui pilote l'imprimante laser ATARI. Il a besoin de 2 méga-octets pour imprimer du texte seul, et de 4 méga-octets si le document à imprimer contient des dessins (voir le chapitre *PERIPHÉRIQUES* : Imprimantes laser).

CONCLUSIONS :

Si vous achetez votre machine dans un but précis, assurez-vous que votre configuration mémoire est suffisante pour les logiciels destinés à votre application.

Si vous possédez déjà un ordinateur et que vous désirez acheter un programme, vérifiez au préalable qu'il supporte correctement votre équipement.

3.2. Les différences de lecteur de disquettes

Il n'existe que 2 types de lecteurs de disquettes : *simple face* (SF) ou *double face* (DF). Le lecteur simple face ne possède qu'une seule tête de lecture/écriture et ne peut donc accéder qu'à un seul côté de la disquette. Un lecteur double face, par contre, possède deux têtes et a donc accès simultanément aux deux faces de la disquettes.

Examinons les répercussions de cette différence mécanique. S'il est clair et net qu'un lecteur, quel qu'il soit, peut lire, écrire et formater les disquettes de son propre type, qu'en est-il des mélanges ?

⇒ Lecture

Un lecteur double face est capable de lire une disquette simple face.

Un lecteur simple face ne sait pas lire une disquette double face, même pas à moitié.

⇒ Ecriture

Un lecteur double face est capable d'écrire sur une disquette simple face, qui pourra être ensuite relue par un lecteur simple face.

Un lecteur simple face ne sait pas écrire sur une disquette double face, même sur un seul côté.

⇒ Formatage

Le formatage est le traitement préalable qu'il faut faire subir à toute disquette vierge pour la rendre utilisable par l'ordinateur.

Un lecteur double face peut formater une disquette au format simple face. Elle sera alors utilisable par un lecteur simple face.

Un lecteur simple face ne peut formater une disquette au format double face, même incomplètement.

CONCLUSION :

Si vous devez échanger un fichier avec un possesseur de lecteur double face, employez une disquette simple face.

☞ Remarques :

✓ Les raisons de ces incompatibilités apparaîtront dans le chapitre consacré à l'*Operating System* (OS).

✓ Tous les logiciels vendus dans le commerce le sont sur des disquettes simple face. Certains offrent parfois le choix entre les deux formats, mais si rien n'est précisé, c'est qu'ils sont fournis sur support simple face. Les utilisateurs de 520 ST ne doivent donc pas se faire de souci pour cela.

✓ Il est possible d'utiliser, à côté ou en remplacement du lecteur 3"1/2 un lecteur 5"1/4. Tout ce qui a été dit jusqu'ici reste valable : le système ne fait pas de différence entre les deux tailles de disquettes; elles sont gérées de la même manière. Ce type de disquette permet, moyennant une certaine discipline, d'échanger des fichiers avec les ordinateurs compatibles IBM. Ce sujet est traité ultérieurement (voir le chapitre *UTILITAIRES* : Emulateur IBM, ainsi que le chapitre consacré à l'OS).

3.3. Les différences du nombre de lecteurs

Bon nombre d'applications - la PAO par exemple - exigent de pouvoir accéder directement à plus de 360 Ko de données. Leur utilisation sur un 520 muni d'un lecteur simple face impose à l'utilisateur de trop fréquents changements de disquettes. Cela conduit généralement à un phénomène plus ou moins prononcé d'impatience que les initiés appellent le "*syndrome du disque-jockey*".

Pour remédier à ce problème, une seule solution: utiliser un deuxième lecteur de disquettes. Il contient généralement les données de l'utilisateur tandis que les programmes et données propres à l'application occupent le premier lecteur.

Pour le possesseur de 1040, la situation est différente. Son unique lecteur de 720 Ko suffit à la quasi-totalité des applications. Sa vaste mémoire lui permet de plus d'en réserver une partie pour un disque virtuel, où il placera les fichiers les plus utilisés.

Que faire lorsqu'une application exige la présence d'un second lecteur que l'on ne possède pas ? La réponse est simple : rien.

Le système gère automatiquement ce cas en créant virtuellement ce second lecteur. Chaque fois que le programme essaiera de lire la disquette placée dans le lecteur "B" - qui n'existe pas -, le GEM reprend le contrôle et affiche un message demandant de placer la disquette en question dans le premier et unique lecteur.



Insérez la disquette B: dans le lecteur A:.

CONFIRMER

S'il - le programme - essaie par la suite d'accéder à la disquette qui devrait normalement se trouver dans le lecteur "A", GEM reprend la main et demande de la replacer dans le lecteur.



Insérez la disquette A: dans le lecteur A:.

CONFIRMER

Toute application prévue pour tourner sur un système équipé de deux lecteurs fonctionne donc correctement avec un seul.

3.4. Les différences d'écran

Il existe pour la gamme ST deux types d'écrans : monochrome (noir et blanc) ou couleur. Il sont totalement incompatibles: *l'écran monochrome ne peut pas simuler les différentes couleurs par des teintes de gris, et l'écran couleur ne peut*

pas afficher d'image destinée à un moniteur monochrome. Cela pourrait être une source de problèmes majeure mais, grâce à GEM, il n'y en a presque pas. On peut diviser, d'un point de vue écran, les programmes en 3 catégories, suivant qu'ils nécessitent un moniteur :

- 1. couleur ou monochrome, au choix
- 2. exclusivement monochrome
- 3. exclusivement couleur

1. Couleur ou monochrome ?

Tous les programmes dont le fonctionnement ne dépend pas du type d'affichage fonctionnent indifféremment dans les deux modes. Ce bonus - pour l'utilisateur - ne demande aux programmeurs aucun effort supplémentaire car GEM, d'origine sur les ST, leur permet d'écrire un programme sans se soucier du type d'écran employé par l'utilisateur.

Figurent ici tous les programmes qui ne sont pas dans les deux autres catégories, c'est-à-dire une très grande majorité d'applications: traitements de texte, langage de programmation, utilitaires, base de données, etc.

2. Exclusivement monochrome

Les programmes exclusivement monochrome le sont parce qu'ils ont un impérieux besoin d'une finesse d'affichage maximale que ne saurait leur accorder la couleur (en mode couleur un "point" couvre 2 ou 4 "points" noir et blanc; on ne peut donc en afficher que 2 ou 4 fois moins sur un écran, d'où une perte de finesse).

Certains logiciels de dessin technique ne travaillent que dans ce mode.

3. Exclusivement couleur

Cette catégorie se limite à deux types de logiciels :

a. Les jeux : la quasi totalité des jeux requiert l'écran couleur.

b. Les logiciels de dessin en couleur.

CONCLUSION :

Si votre seul but est de jouer avec votre ordinateur, ou bien d'être un nouveau Picasso mâtiné de ST, le choix du moniteur couleur s'impose.

Si votre choix est mitigé, rappelez-vous qu'il suffit d'un simple branchement pour changer de type d'écran, et que rien ne vous empêche d'employer les deux. Vous pouvez encore remplacer le moniteur couleur par une télévision à prise Péritel ou un téléviseur courant si votre ordinateur est muni d'un modulateur.

☛ Remarques :

✓ Pour un travail prolongé, sur un traitement de texte par exemple, c'est le moniteur monochrome qui se révèle le plus confortable à l'usage et à la longue le moins fatigant pour les yeux.

✓ Le mode couleur est divisé en 2 sous-modes : moyenne et basse résolution, suivant le nombre de couleurs distinctes simultanément présentes à l'écran et la définition de l'image (voir le chapitre consacré au *GRAPHISME*). Tout moniteur couleur peut travailler indifféremment dans ces 2 sous-modes.

✓ Lorsque le GEM n'est pas capable d'assumer la différence d'écran d'une manière assez performante - rapidité - mais que les programmeurs ne veulent léser personne, ils fournissent parfois les deux versions - couleur et monochrome - sur la même disquette. Ces programmes se reconnaissent aisément à leur nom, car ils diffèrent seulement par le suffixe - "M" ou "C" - suivant qu'ils fonctionnent sur écran couleur ou monochrome.

Exemples : "gameM.prg"
 "gameC.prg"

✓ Une autre possibilité consiste en un programme monobloc, lequel possède cependant un double jeu de routines d'affichage, un pour la couleur et l'autre pour le monochrome. Lors de son lancement, il détecte le type du moniteur et sélectionne le jeu adapté. Cela simplifie légèrement la tâche de l'utilisateur, mais le programme occupera en mémoire une place plus importante.

✓ Il est possible, dans certains cas, d'afficher les 2 types d'affichage sur un seul écran (voir le chapitre consacré aux *UTILITAIRES* : Emulateur écran et voir également le chapitre consacré aux *PERIPHERIQUES*: Extension. Ecran 3 modes).

3.5. Les différences nationales de l'OS

Suivant le pays et la langue de l'utilisateur, deux caractéristiques de l'ordinateur peuvent changer :

- 1 Le clavier
2. Les messages du GEM.

1. Le clavier

Il existe de nombreuses versions nationales du clavier ATARI parmi lesquelles on trouve :

QWERTY	anglophone
AZERTY	francophone
QWERTZ	germanophone

⇒ **Avantage :**

L'utilisateur peut conserver les automatismes acquis sur les machines à écrire de son pays.

⇒ **Désavantage :**

Certains programmes exploitent le clavier d'une façon non réglementaire, en court-circuitant l'OS. Cela produit parfois un comportement aberrant. Prenons un exemple parmi tant d'autres.

"Le Rédacteur", un logiciel de traitement de texte français supporte sans problème apparent son utilisation sur un clavier QWERTY. Il reconnaît directement toutes les lettres et chiffres du clavier, en majuscules ou minuscules, et ce malgré la localisation différente de certains d'entre eux (a,z,w,m,...). Mais là où le bât blesse, c'est lorsqu'ils sont employés en combinaisons avec des touches spéciales. [CTRL Q] par exemple, - la touche "Q" pressée en même temps que la touche "CTRL" - est confondue avec [CTRL A]. Ce problème n'est pas mineur car ce programme utilise beaucoup de combinaisons de ce type.

Ce genre de mésaventures peut aussi vous arriver sur votre clavier AZERTY si vous utilisez un logiciel étranger non ou mal traduit. Prudence donc.

☉ **Remarque :**

Si un utilisateur francophone veut utiliser un clavier QWERTY, il existe de nombreux

programmes utilitaires qui permettent de le reconfigurer pour lui donner accès à nos caractères spécifiques : accents, cédille, tréma. Cette facilité est intégrée d'origine dans la plupart des traitements de textes.

2. Les messages du GEM

Les messages du GEM sont ceux qu'il vous envoie quand, par exemple, vous lui demandez de détruire un fichier protégé ou que vous avez oublié de mettre une disquette dans le lecteur.

A chaque langue supportée correspond une version traduite de ces messages. Cette différence n'est source d'aucune incompatibilité.

CONCLUSION :

Si vous achetez un logiciel d'origine étrangère, assurez-vous qu'il est traduit ou au moins totalement compatible avec votre clavier. Sans cela, un produit merveilleux pourrait se révéler inutilisable.

☞ Remarque :

Les problèmes de compatibilité évoqués ici ne peuvent se produire qu'avec les logiciels qui font un usage intensif et "pointu" du clavier. N'appartiennent pas à cette catégorie, entre autres, les logiciels de dessins, les utilitaires ou, plus généralement, tous ceux qui ne s'en servent que pour la saisie simple de texte, en évitant toute combinaison de touches litigieuse.

3.6. Les versions successives de l'OS

Avant d'en parler plus en profondeur dans le chapitre consacré à l'OS, disons simplement pour le profane qu'il s'agit d'un programme système résidant en permanence dans la machine et qui la gère. C'est lui qui permet l'impression de texte, le formatage de disquettes, la destruction de fichiers, etc.

Mises à part les préversions fournies sur disquettes dans les premiers temps, il existe deux versions successives, en ROM, du système d'exploitation. La première a équipé les modèles 260, 520 et 1040 jusqu'à la sortie du MEGA ST. Comme tout programme, elle contenait quelques erreurs - pas très graves - qui ont été corrigées pour fournir la *version 2*. Cette nouvelle version - on dit aussi les "nouvelles ROM" - a équipé le MEGA ST dès sa sortie ainsi que tous les 520 et 1040 produits par la suite.

Il s'est avéré que certains logiciels du commerce qui fonctionnaient correctement avec l'ancien OS ne tournaient plus avec le nouveau. La cause en revenait aux programmeurs qui n'avaient pas respecté les consignes données par ATARI pour permettre à leurs programmes de supporter les futures versions de l'OS. Les versions suivantes de ces programmes ont été corrigées, et aucun de ceux qui arrivent actuellement sur le marché ne connaît plus ce problème. Quand l'adaptation s'avère impossible, pour des raisons de performance, le programme est vendu en deux versions: une pour les nouvelles, une pour les anciennes ROM. Mais ce cas se présente rarement.

S'il vous arrivait de placer des nouvelles ROM dans votre machine, sachez que les fabricants de logiciels procèdent au remplacement, moyennant léger

supplément, des programmes litigieux qui ne tourneraient plus sur votre machine ainsi équipée.

3.7. L'accès au bus

Comme nous l'avons vu plus haut, l'accès au bus permet entre autres choses de raccorder au MEGA ST un écran géant. Il est évident que les programmes qui le réclament ne fonctionneront pas sur un 1040 ou un 520, puisqu'ils n'en sont pas équipés.

Ceci clôture l'étude des différences entre les modèles de la gamme ST.

4. L'homogénéité et ses avantages

On peut conclure que les possesseurs d'ATARI ST forment une population relativement homogène en ce qui concerne leur équipement. Cela facilite le travail des programmeurs - ils ne doivent écrire qu'une version de leur programme - et le risque qu'un programme qui vous convient n'accepte pas votre configuration est presque nul.

OUVERTURE MS-DOS

Même si l'utilisateur moyen ne s'en rend pas compte, le système d'exploitation de l'ATARI ST a de nombreuses similitudes avec le MS-DOS, le système d'exploitation universel des ordinateurs IBM et compatibles.

Les lecteurs de disquettes que nous employons sont identiques et les conventions employées pour écrire et formater ces disquettes sont (presque) les mêmes.

Le résultat de ces similitudes est que l'on peut échanger des données - du texte par exemple - sur disquettes avec le possesseur d'un ordinateur IBM. Quelques précautions initiales sont cependant requises. Elles seront traitées plus loin, dans le chapitre consacré aux *UTILITAIRES* : Emulateur IBM.

LE SYSTEME d'EXPLOITATION



KÉSAKO ?

⇒ ***L'OS est l'âme de l'ordinateur.***

Sans lui, il ne serait qu'un amas inerte de circuits inutilisables. L'écran resterait muet et le clavier insensible à nos plus ardentes sollicitations.

⇒ ***L'OS est un programme.***

Ou plutôt un ensemble de programmes. Chacun d'eux s'occupe d'une tâche spécifique telle que gérer le clavier, l'écran, la souris, le lecteur de disquettes, etc. Pour cela, ils communiquent directement avec les processeurs périphériques. L'OS assure aussi la cohérence entre toutes ses parties.

⇒ ***L'OS cache le hardware.***

Un logiciel qui veut utiliser les ressources matérielles du ST n'a pas à se soucier de leur réalité physique - exemple: comment imprimer du texte à l'écran - mais doit simplement demander à l'OS de le faire pour lui. Si une des caractéristiques de la machine est modifiée dans une version ultérieure, la nouvelle version de l'OS rendra cette modification invisible aux anciens programmes qui continueront à s'exécuter correctement, sans s'apercevoir de rien.

SES ROLES :

Pour chaque périphérique, il offre au programmeur une batterie de routines. Retenons les principales :

Périphérique concerné	Routines
Lecteur	Lire / écrire des données sur disquette. Formater une disquette. Effacer, renommer, copier des fichiers. Organiser les données sur la disquette.
Clavier	Saisir un caractère.
Souris	Saisir la position de la souris.
Ecran	Ecrire, effacer, dessiner .
Imprimante	Envoyer du texte ou des commandes.
Modem	Saisir / envoyer des caractères au modem.

De plus, il assure la cohésion de l'ensemble.

☞ Remarques

✓ A l'origine, l'ATARI ST devait être équipé du système d'exploitation CP/M68 de Digital Research (DRI). Il fut finalement remplacé, principalement pour des raisons de performances, par un produit maison qui porte le doux nom de T.O.S. (*Tramiel Operating System*).

✓ Contrairement à la majorité des ordinateurs, un ATARI ST peut s'utiliser sans posséder aucune connaissance de l'OS. Ceci est rendu possible par l'interface graphique du GEM qui prend en main l'utilisateur dès le début de sa session de travail.

✓ Bien qu'étant installé en ROM, le système d'exploitation n'est pas figé. Si une nouvelle version est disponible en ROM, il suffit de procéder à l'échange pour faire évoluer votre installation.

GESTION DU DISQUE :

Une étude approfondie de l'OS sort du cadre cet ouvrage. Il est cependant intéressant d'avoir une idée de la manière dont il gère disquette et disque dur.

Lorsque vous déballez une disquette neuve, elle est impropre à la consommation. Pour la rendre utilisable, elle doit être formatée par l'OS.

➔ **Le formatage.**

Le formatage prépare la disquette à recevoir des informations. Chaque face de sa surface magnétisable est divisée en 80 pistes concentriques de 9 secteurs chacune. Une *piste* est une portion circulaire de la disquette. Elle sont numérotées de 0 à 79, de l'extérieur vers l'intérieur. Un *secteur* est une portion contiguë d'une piste. Il héberge 512 octets d'information.

Après formatage, une disquette simple face a donc la capacité ci-après:

$$\begin{aligned} & 80 \text{ (pistes)} \times 9 \text{ (secteurs)} \times 512 \text{ octets} \\ & = 368.640 \text{ octets} \\ & = 360 \times 1024 \text{ octets} \\ & = 360 \times 210 \text{ octets} \\ & = 360 \text{ Ko (kilo-octets)} \end{aligned}$$

Ces 360 Ko ne sont pas à l'entière disposition de l'utilisateur. Le système d'exploitation s'en réserve une partie pour la gestion future de la disquette. Il y placera entre autre le nom, la taille, la localisation et la date de dernière modification de tous les fichiers de données que l'on y aura stocké. Sur une disquette simple face, il utilise à cet usage 22 secteurs, ce qui laisse à l'utilisateur 349 Ko pour y placer ses données.

Une disquette *double face* est une disquette dont on emploie les deux faces du support magnétisable. Chacune d'elles contient 80 pistes de 9 secteurs, ce qui assure une capacité totale de 720 Ko.

Sur l'ATARI ST, le formatage d'une disquette se fait à partir du *Bureau du GEM*, en sélectionnant avec la souris l'option "Format" du menu "Fichier" (voir le chapitre consacré à GEM).

⇒ Un fichier

Qu'il corresponde à un programme, du texte, des graphiques ou à tout autre type de données, un fichier est toujours stocké d'une façon identique. Il est découpé - logiquement - par l'OS en *morceaux de 1024 octets* qui sont installés dans les secteurs libres (pas encore occupés) de la disquette. Ils peuvent être dispersés aux quatre coins - c'est une façon de parler - de la disquette. L'OS retient dans les secteurs réservés

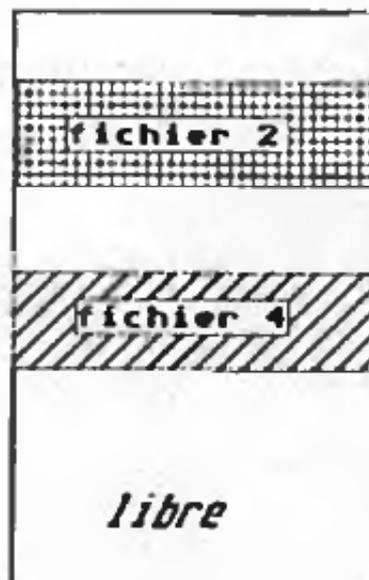
au système où et dans quel ordre sont copiés les différents morceaux du fichier.

➔ **Répartition des morceaux**

Sur une disquette fraîchement formatée, tous les secteurs réservés à l'utilisateur sont inoccupés. Imaginons que l'on désire y sauver quatre petits fichiers. L'OS part à la recherche de place libre et les installe l'un derrière l'autre. Le schéma d'occupation de la disquette est le suivant :



Si nous effaçons les fichiers F 2 et F 4, on obtient le résultat suivant :



⇒ **La fragmentation.**

Lorsqu'on demande par la suite à l'OS de copier un nouveau fichier, il commence sa recherche d'espace libre par le début de la disquette. Dès qu'il trouve de la place, il la remplit avec le fichier à sauver. Si celui-ci est plus grand que la taille disponible, l'OS poursuit sa recherche et procède de la sorte avec le reste du fichier.

Copions un grand fichier : F5



Nous voyons qu'il n'est plus stocké d'une manière contiguë, mais qu'il est "*fragmenté*". Lorsqu'une disquette est employée longtemps sans être reformatée, il n'est pas rare qu'un - ou des - fichiers y soient dispersés en plusieurs dizaines de morceaux.

La conséquence directe de cette fragmentation est l'allongement de la durée du chargement du fichier. La tête de lecture du lecteur doit en effet se déplacer d'une piste à l'autre, à la recherche des fragments dispersés du fichier et attendre qu'elle passe au-dessus des secteurs qu'il occupe. Le même raisonnement s'applique, dans l'autre sens, à la sauvegarde du fichier sur la disquette.

⇒ **Le pire des cas de fragmentation.**

L'OS alloue l'espace de la disquette par "*cluster*", c'est-à-dire par paire de secteurs consécutifs (qui se suivent sur la même piste ou le dernier d'une piste et le premier de sa voisine). Ce cluster ne peut être occupé que par un fichier à la fois. Si sa taille n'est pas un multiple exact de 1024, le dernier cluster qu'il occupera ne le sera que partiellement. Cela veut dire que des fichiers de 1, 2, 3 ou 1024 octets occupent le même espace sur le disque, à savoir deux secteurs, c'est-à-dire 1 Ko

Dans le pire des cas de fragmentation, un fichier de 60 kilo-octets sera dispersé sur 60 pistes différentes, occupant deux secteurs sur chacune. On comprend dès lors le ralentissement de son chargement.

⇒ **Remèdes anti-fragmentation**

Il est impossible de prévenir la fragmentation, on peut juste la supprimer une fois qu'elle s'est installée. Pour cela, deux moyens: le *back-up* ou le *défragmenteur*.

▲ **Le Back-up**

L'enfance de l'art: il suffit de copier intégralement le contenu de la disquette fragmentée sur une deuxième disquette, vide ou que l'on vient de formater. Les fichiers dispersés sur la première seront recopiés à la suite les uns des autres sur la seconde.

Remarque : ce procédé est gratuit, optimum et pleinement fiable. Il demande juste un peu de temps et une disquette supplémentaire.

▲ **Le Défragmenteur**

Il s'agit d'un programme qui réarrange le contenu de la disquette. Plusieurs méthodes existent, mais elles ont toutes un point commun : si un problème se

produit pendant l'opération, vous risquez de perdre tout ou partie de vos fichiers. Leur usage se justifie surtout pour la défragmentation du disque dur, le back-up complet sur disquette étant particulièrement long et inconfortable.

➔ ***"The forty folders bug"***

Parmi les bugs de l'OS, il en est un qui gêne plus particulièrement les possesseurs de disque dur : la **limite des 40 dossiers**. Expliquons. Si vous parcourez plus de 40 dossiers lors d'une même session de travail, l'OS se plante lamentablement, mais sans rien dire. Cela se manifeste de deux manières. D'une part, lorsque vous demandez le chargement et l'exécution d'un programme, il vous envoie un message d'alerte prétendant que vous ne disposez plus d'assez de mémoire pour cette application. D'autre part, une partie de votre disque dur peut devenir invisible, comme si vous aviez tout effacé. Pas de panique. Vous n'avez rien perdu. Quand un de ces deux symptômes se déclare, il n'y a qu'une chose à faire : réinitialiser votre système. Les fichiers disparus réapparaîtront aussitôt.

Remède

Il suffit de placer, dans le fichier AUTO de votre disquette de chargement, le "patch" - petit programme qui corrige une erreur - ad hoc. Vous pouvez l'obtenir facilement chez votre vendeur ou dans un club d'utilisateurs.

Notons que cette erreur a été corrigée dans la seconde version de l'OS et qu'elle n'atteint donc que les "vieux" ST.

☞ **Remarques :**

✓ Tout ce que nous venons de voir ici s'applique aussi au disque dur, si ce n'est qu'il nécessite d'avoir l'OS en ROM.

✓ Le formatage d'une disquette se fait à partir du Bureau du GEM, en sélectionnant avec la souris l'option "Format" du menu "Fichier" (voir le chapitre consacré au GEM).

⇒ **Format étendu :**

Le T.O.S. accepte que l'on prenne quelques libertés avec son format standard : on peut mettre 10 secteurs par pistes et plus de 80 pistes par face. Selon la qualité des disquettes, on peut utiliser 82, 83 voire 84 pistes.

Faisons un bref calcul :

$$82 \times 10 \times 512 = 410 \text{ Ko}$$

par face au lieu de 360 avec le format standard.

Ajoutons qu'une disquette formatée en mode étendu sur votre machine reste exploitable par tous les autres ST.

AUTRES OS :

Il est possible de remplacer le TOS par un autre système d'exploitation. Celui-ci devra être copié en RAM à partir d'une disquette, comme avec les premières versions du TOS.

☛ Conséquences:

Toutes les applications conçues pour tourner avec le TOS deviennent inutilisables. De même que le GEM. Cela implique qu'il faut réécrire de nouveaux programmes.

On peut, dans certains cas, récupérer ceux écrits pour d'autres machines 68000 dotées de cet OS.

Un nouvel OS doit donc être accompagné d'une gamme de produits - éditeur, compilateur, utilitaires - le rendant effectif.

☛ Pourquoi un autre OS ?

Les autres OS disponibles sur ATARI ST offrent tous une potentialité supplémentaire que le TOS est incapable de fournir : ils sont multi-tâches et/ou multi-utilisateurs.

OS multi-tâche : (multi-tasking)

L'utilisateur peut faire s'exécuter en même temps plusieurs tâches - programmes - présents en mémoire. Il s'agit de pseudo-parallélisme : chaque programme reçoit de l'OS l'autorisation de disposer de la machine pendant une fraction de temps - quelques millisecondes - après quoi un autre prend sa place.

OS multi-utilisateur : (multi-users)

Plusieurs personnes peuvent travailler simultanément sur la mémoire centrale d'une seule machine (le poste maître). Chaque utilisateur doit évidemment disposer d'un poste de travail (clavier + écran) individuel.

La machine centrale est appelée le *poste maître*. Les autres sont les terminaux, ou *postes*

esclaves. C'est pour cela que l'on parle aussi de système multi-postes.

Remarque : normalement, seul le poste maître dispose d'une mémoire propre. C'est là que sont chargés et exécutés tous les programmes employés par les différents utilisateurs simultanés. La situation peut varier légèrement chez ATARI, un petit modèle de la gamme - 520 ou un 1040 - pouvant servir de terminal. Il est alors possible de l'utiliser de deux manières : en mode esclave, comme un terminal relié au poste central ou classiquement, comme un ATARI ST normal, avec GEM et le TOS.

Un système multi-utilisateur a plusieurs avantages sur un ensemble d'ordinateurs isolés. Les plus importants sont :

- ✓ le partage du matériel;
- ✓ le partage de données;
- ✓ l'unicité du logiciel.

○ **Partage du matériel :**

Il suffit d'un exemplaire de chaque périphérique - imprimante laser, disque dur - pour tout le système, mais ils sont directement accessibles par chaque utilisateur du système. Ces périphériques sont reliés au poste maître qui s'occupe de leur gestion. Cela s'avère nécessaire pour éviter les conflits de simultanéité : deux ou plusieurs utilisateurs envoyant un texte vers l'imprimante ou voulant modifier, par exemple, un fichier en même temps.

○ **Partage de données :**

Le disque dur étant utilisable par tous, les données qu'il contient le sont aussi. Si elles représentent l'état d'un stock de véhicules, plusieurs personnes pourront le consulter en même temps. Lorsqu'il est modifié par un des utilisateurs (un seul à

la fois), tous peuvent s'en apercevoir instantanément. Cela évite de vendre deux fois la même voiture.

○ **Unicité du logiciel :**

Si plusieurs personnes emploient le même programme en même temps, il est inutile de le copier plusieurs fois dans la mémoire du poste maître. On peut concevoir ce logiciel pour qu'un seul exemplaire suffise et puisse être employé par plusieurs utilisateurs simultanément. On dit alors que le programme est *réentrant*.

OS multi-tâche et/ou multi-utilisateur :

Ces deux systèmes peuvent être combinés de quatre manières différentes :

↳ Mono-tâche, mono-utilisateur.

Le ST d'origine. Un seul poste sur lequel tourne un seul programme.

↳ Multi-tâche, mono-utilisateur.

Un seul poste sur lequel peuvent tourner plusieurs applications en même temps.

↳ Multi-utilisateur, mono-tâche.

Plusieurs postes sont raccordés à un poste maître. Chacun d'eux ne peut lancer qu'un seul programme à la fois.

↳ Multi-utilisateur, multi-tâche.

Plusieurs postes sur chacun desquels peuvent être exploités simultanément plusieurs programmes.

DES NOMS, DES NOMS :

Sur le marché des OS multi-tâches et/ou multi-utilisateurs sont disponibles, entre autre, les programmes suivants :

⇒ OS9

⇒ IDRIS :

- multi-utilisateur;
- compatible avec le 1984/usr/group Unix;
- langages disponibles : Pascal, C, etc.

⇒ MIRAGE :

- multi-utilisateur et multi-tâche;
- disponible en ROM;
- du 520 au MEGA;
- langages disponibles : Assembleur, Basic, C, Fortran, Pascal, Lisp;
- nombreux programmes existant, dont OMNI-3, un SGDB.

☞ Remarques :

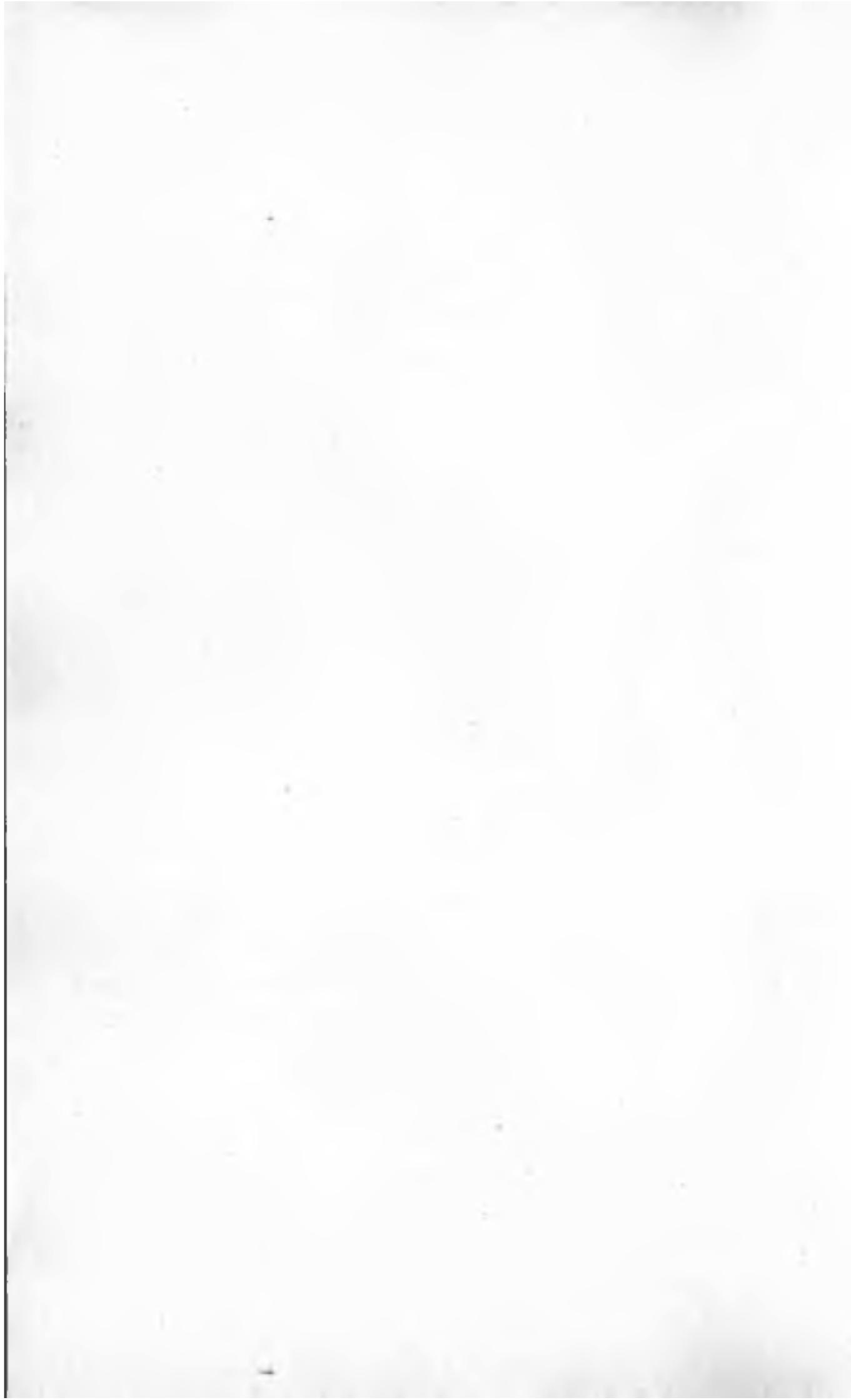
✓ Certains utilitaires, tournant sous TOS et GEM, permettent à plusieurs logiciels de résider en même temps dans la mémoire. Il ne rendent nullement le TOS multi-tâche car il n'y qu'une seule application qui puisse fonctionner à la fois (voir le chapitre consacré aux *UTILITAIRES* : Corésidence).

✓ L'ATARI ST est livré d'origine avec un émulateur VT 50, ce qui signifie qu'il peut remplacer un terminal adapté à la norme VT 50. Il suffit de le connecter à la place de ce dernier et de lancer l'émulateur. D'autres

programmes existent qui le transforment en terminal ASCII, VT 100, Minitel, Viditel, etc. Il peut donc, pour un coût modique, être connecté aux "gros" systèmes.

✓ Le GEM permet une certaine forme de "multi-tasking", mais elle est restreinte à une certaine classe de programmes système : les accessoires. Cette particularité est détaillée dans le chapitre consacré au GEM.

GEM



☛ NON

L'ATARI ST - alias "*Jackintosh*" - n'est pas une pâle copie du Macintosh d'Apple. Si leurs interfaces graphiques se ressemblent tant, c'est parce qu'ils utilisent tous deux une même idée venue d'ailleurs : le W.I.M.P., de Xerox.

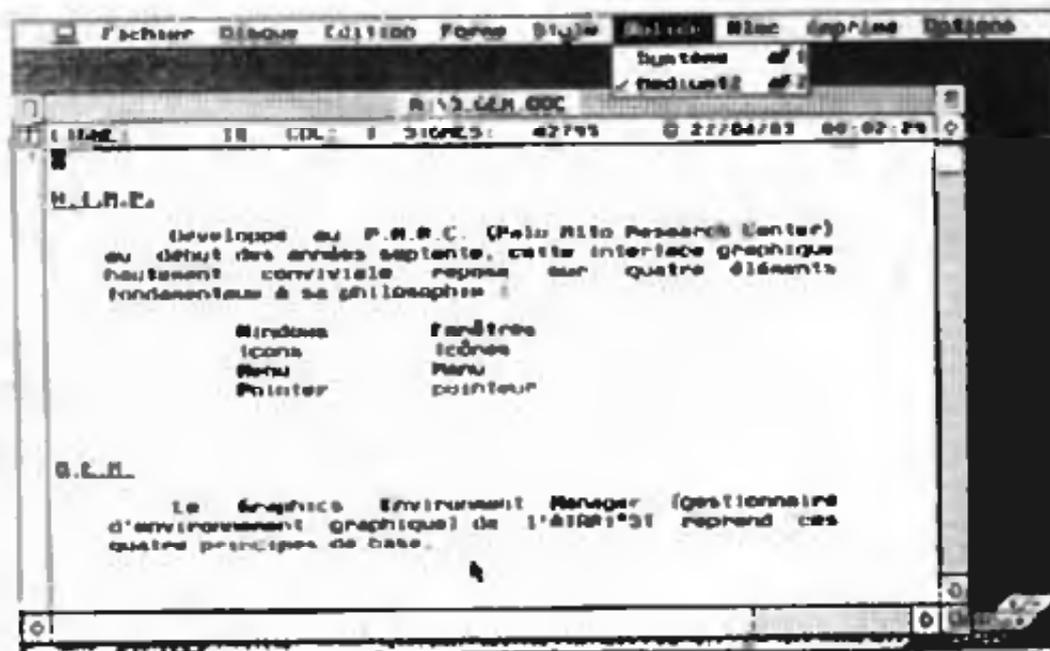
☛ W.I.M.P.

Développé au P.A.R.C. (Palo Alto Research Center), au début des années soixante-dix, cette interface graphique hautement conviviale repose sur quatre éléments fondamentaux à sa philosophie :

☛ Windows	Fenêtres
☛ Icons	Icônes
☛ Menu	Menu
☛ Pointer	Pointeur

☛ G.E.M.

Le *Graphics Environment Manager* (gestionnaire d'environnement graphique) de l'ATARI ST reprend ces quatre principes de base.



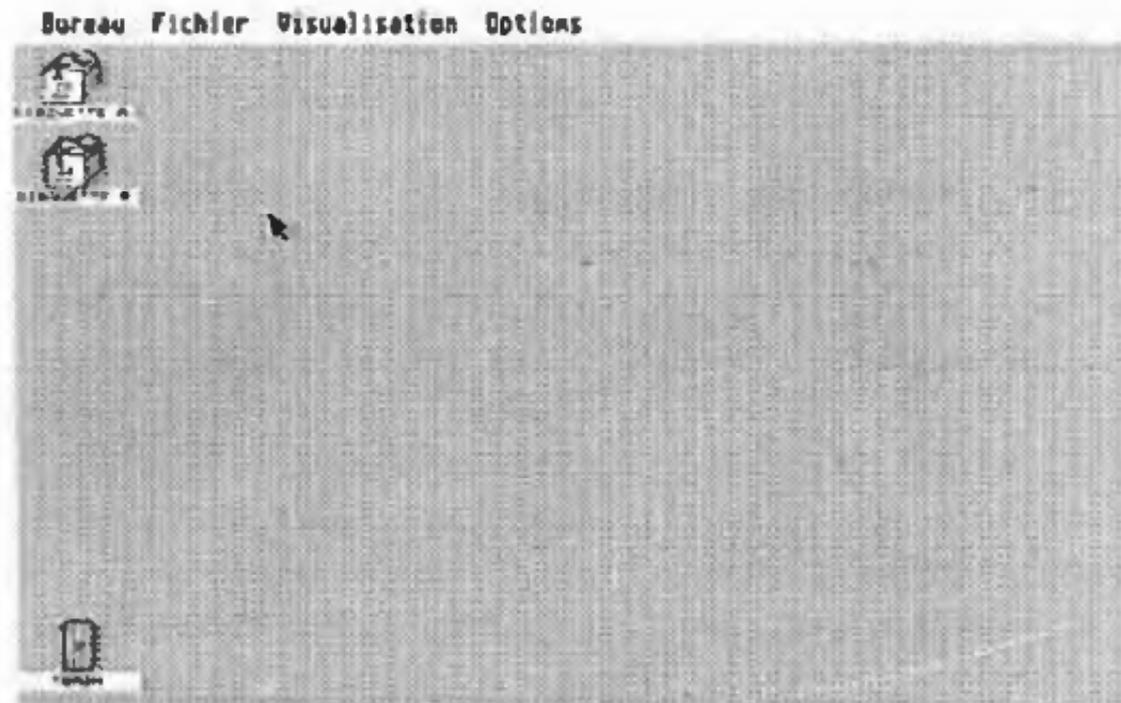
Examinons dans un premier temps le GEM comme il est perçu par l'utilisateur avant de parler des services qu'il offre au programmeur et de la possibilité de se passer du GEM sur le ST.

1. GEM pour l'utilisateur

Où l'on apprend tout du *Bureau* et du *Boot*.

Premiers contacts :

A la mise sous tension de l'ordinateur, l'écran présente l'aspect suivant (version monochrome) :



C'est ce que l'on appelle le *Bureau*, ou *Desktop*. C'est un programme omniprésent qui permet de communiquer avec le système d'exploitation. Les ordres sont envoyés au moyen du *pointeur* (flèche) associé à la *souris*. Leurs effets sont immédiatement répercutés à l'écran s'il y a lieu.

Un bref regard (pas trop bref quand même!) nous permet de distinguer les éléments suivants :

○ Un pointeur

○ Trois icônes :

lecteur A

lecteur B

poubelle

○ Une barre de menus composée de 4 titres :

Desk (Bureau)

File (Fichier)

View (Visualisation)

Options (Options)

☛ **La souris :**

Ses déplacements sur la surface de travail sont répercutés à l'écran par ceux du pointeur (flèche). Les boutons qui ornent son dos permettent deux types d'actions : le *clic* et le *double-clic*. C'est avec délectation que nous emploierons les barbarismes suivant : "cliquer" et "double-cliquer". Précisons, pour les malheureux profanes, qu'un double-clic est formé de deux clics successifs très rapprochés dans le temps sur le même bouton. Sauf spécification contraire, il s'agira toujours d'une action sur le bouton gauche. On emploiera parfois "*sélectionner*" comme synonyme de cliquer. L'objet ainsi sélectionné change d'aspect : il noircit, preuve que l'opération s'est déroulée correctement. On peut aussi "*tirer*" un objet avec la souris : on le sélectionne et, sans relâcher la pression sur le bouton gauche, on déplace le pointeur vers un autre endroit de la fenêtre.

☛ L'aspect du Bureau :

Tout d'abord un conseil de vieux routier : poubelle en bas à gauche, disquettes en haut à droite. Ou l'inverse si vous préférez. Cela vous évitera moult gaffes, aussi douloureuses que faciles à commettre. Personnalisons donc notre bureau.

☛ Déplacement d'une icône :

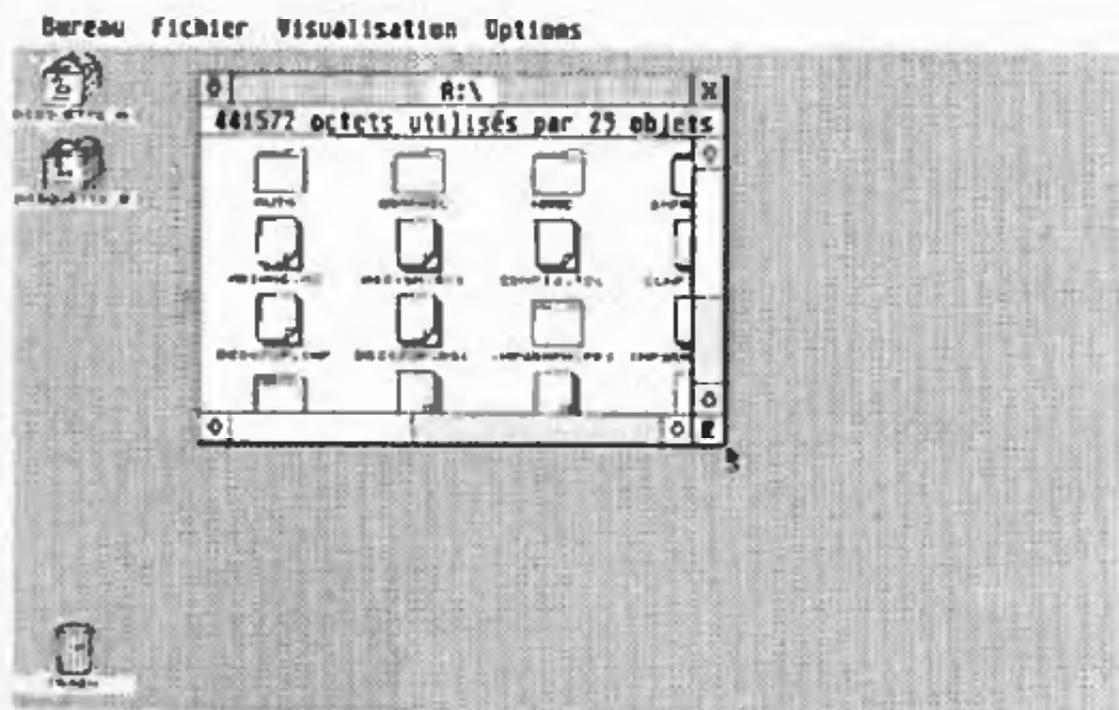
Pas de mystère : cliquez, tirez, lâchez. Expliquons calmement pour ceux qui n'ont pas compris. Amenez le pointeur au dessus de l'icône en question. Cliquez-la discrètement - elle se noircit - et, sans relâcher la pression sur le bouton gauche, tirez-la de force vers l'endroit par vous choisi. Relâchez ensuite le bouton. Vous aurez constaté pendant sa traction que l'icône gardait sa place tandis que son "fantôme" suivait votre pointeur.

☛ Les secrets de l'icône :



Ce symbole est notre seule porte d'accès au lecteur de disquette "A". Le *sésame* : double-cliquez cette icône. Elle se noircit. Le lecteur de disquettes se met à bourdonner pendant que la flèche se transforme en petite abeille.

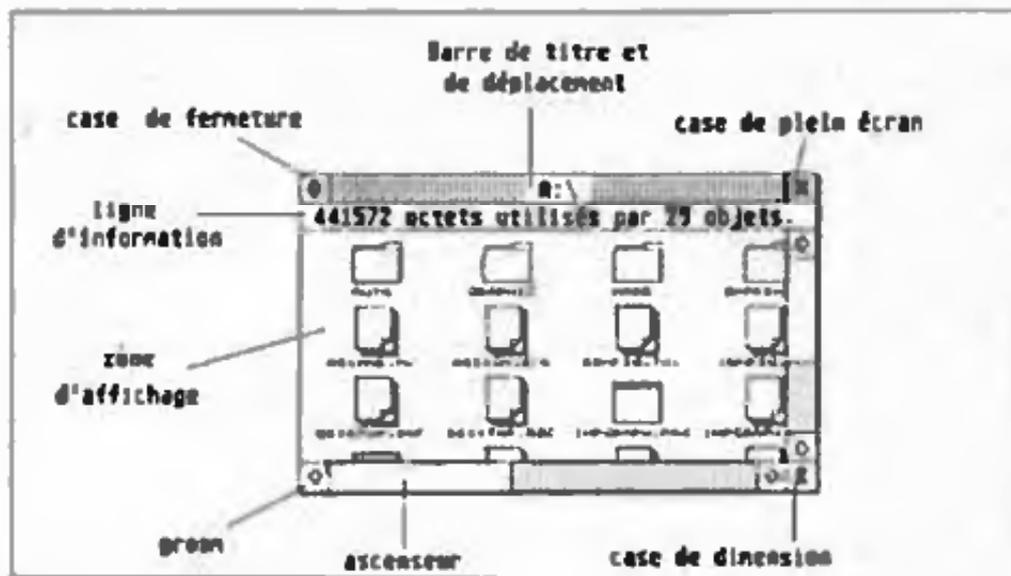
S'ouvre alors une fenêtre :



L'abeille disparaît, la flèche revient, le lecteur se tait et l'icône retrouve sa pâleur originelle. Avant de parler du contenu de la fenêtre, examinons d'abord sa structure point par point.

Première fenêtre :

Elle est constituée d'un cadre intérieur, d'une barre de titre et de déplacement, d'une ligne d'information et d'outils - cases - pour la manipuler.



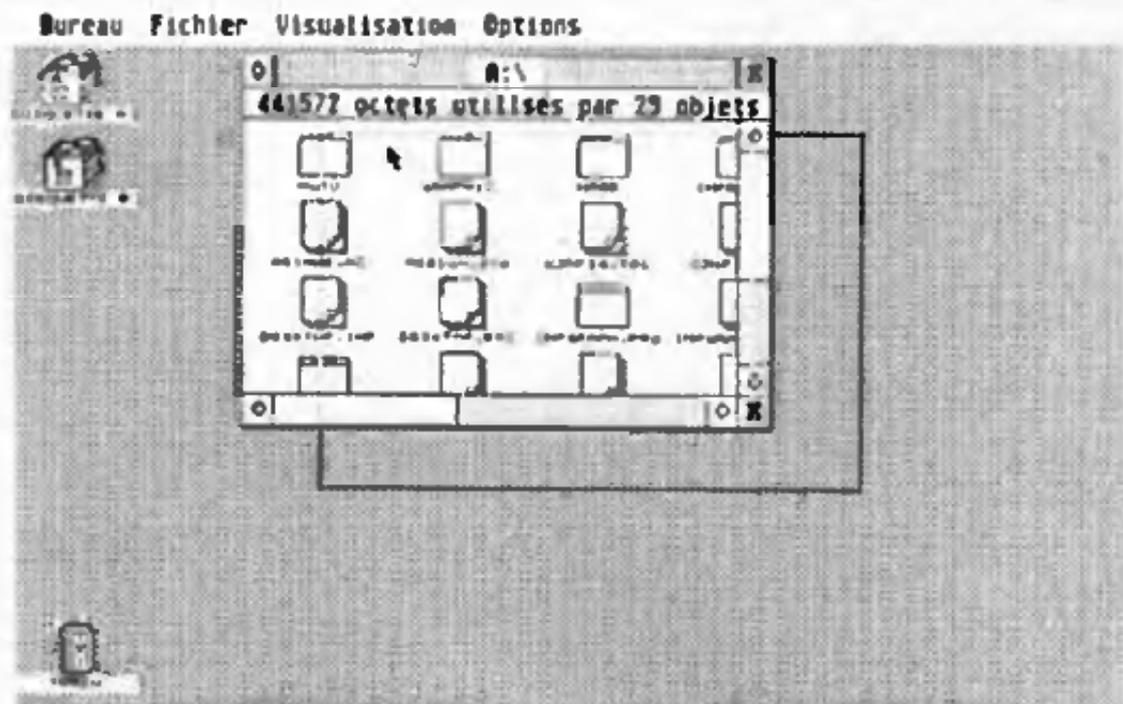
→ Zone de travail

Elle contient le catalogue de la disquette. Nous en parlerons plus tard.

→ La barre de titre et de déplacement

Le *titre* : c'est le nom du lecteur de disquettes.

Le *déplacement de la fenêtre* : il s'effectue comme celui d'une icône. Cliquez la barre de déplacement et tirez la fenêtre vers sa nouvelle position. Relâchez la pression. Pendant l'opération, le fantôme de la fenêtre suit la souris :



⇒ **La case de fermeture**

Il suffit de la cliquer pour fermer - faire disparaître du bureau - la fenêtre.

⇒ **La case de plein écran**

Un premier clic étend la fenêtre sur tout l'écran. Un second lui redonne ses taille et position initiales.

⇒ **La case de dimension**

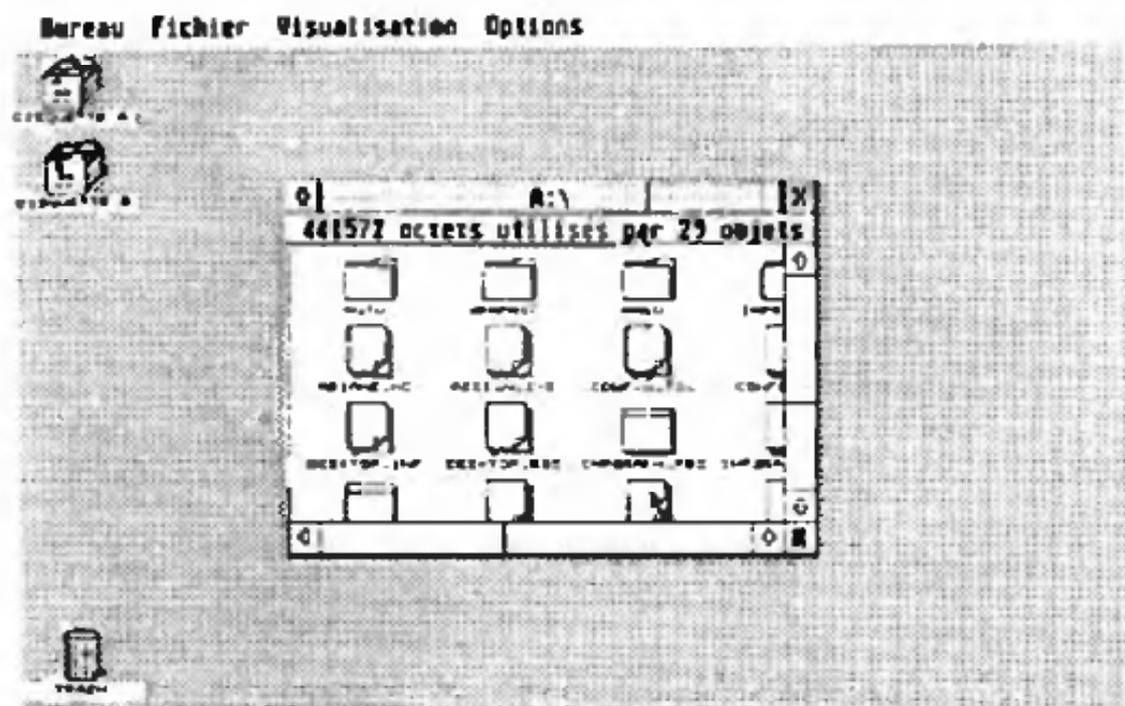
Elle vous permet de modifier la taille de la fenêtre.

1° Cliquez la case

2° Tirez la case

3° Relachez la pression

Pendant l'opération, le fantôme de la fenêtre suit le pointeur et indique la dimension future de celle-ci :

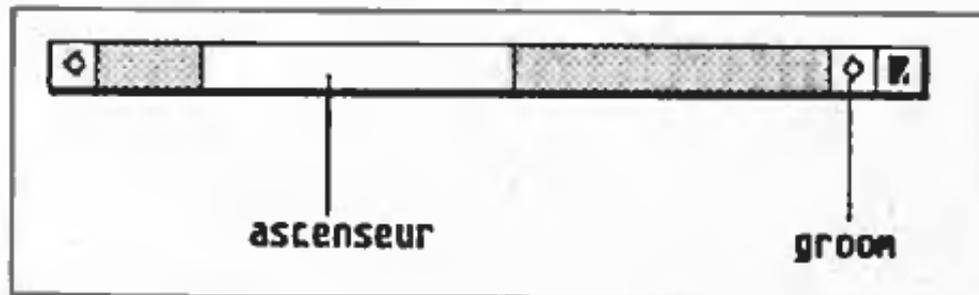


⇒ La ligne d'information

Elle indique le nombre de fichiers - programmes ou données - présents à ce niveau ainsi que la somme de leurs tailles respectives.

Si la fenêtre est trop petite, certaines icônes sont invisibles. Elles se trouvent au-dessus ou en dessous, à droite ou à gauche de la portion d'écran visible dans la fenêtre. Pour voir ces icônes, il suffit de faire glisser la fenêtre dans la direction voulue. C'est le rôle des deux bandes de défilement, verticale et horizontale. Regardons-les de plus près.

⇒ Les ascenseurs et les bandes de défilement



Le rectangle blanc - l'ascenseur - représente ce que vous voyez dans le cadre. Les zones grises représentent, elles, ce que vous ne voyez pas, c'est-à-dire les icônes qui se trouvent hors du cadre. La taille de l'ascenseur dans la bande de défilement est proportionnelle à la surface visible dans la fenêtre.

Comment faire apparaître des objets sortant du cadre ?

En faisant défiler le contenu de la fenêtre. Pour cela, trois possibilités :

1° Ligne ou colonne suivante.

Cliquer le groom horizontal droit (gauche) permet de visualiser la colonne d'icônes cachée suivante (précédente).



2° Cadre suivant.

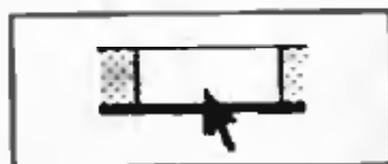
Cliquer la zone grisée à droite (gauche) de l'ascenseur, fait défiler le cadre d'une largeur de fenêtre vers la droite (gauche).



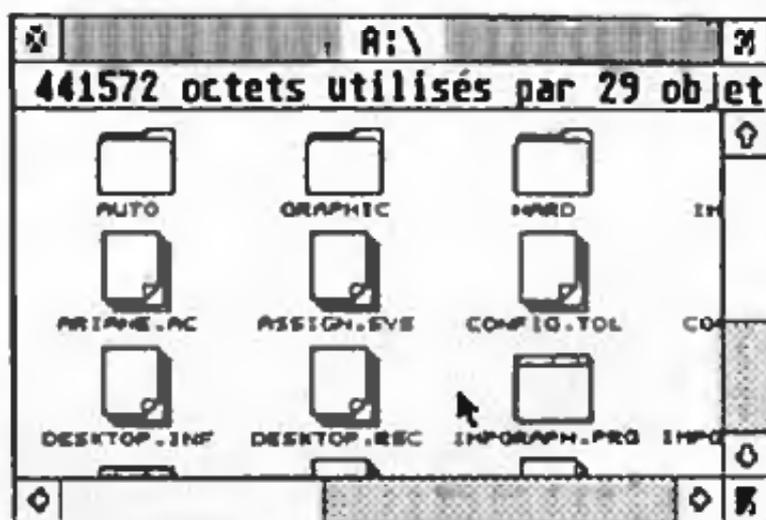
3° Positionnement direct du cadre.

Tirer l'ascenseur permet de positionner directement le cadre sur la surface totale.

La barre de défilement verticale permet de faire défiler le cadre vers le haut ou vers le bas.



Le catalogue (directory) :



On peut distinguer trois types d'icônes:

○ L'icône de programme

Elle est associée à un programme exécutable. Pour le lancer, double-cliquez son icône. Suivant le suffixe, on distingue trois types de programmes :



- ✓ TOS
- ✓ TTP (APP)
- ✓ PRG

⇒ TOS

Programme n'employant pas l'interface graphique du GEM, mais uniquement le système d'exploitation (le TOS). Il n'utilise donc pas d'icône, de fenêtre ni de barre de menu.

⇒ TTP (Tos Takes Parameters)

Comme le TOS, il n'utilise pas le GEM, mais il demande qu'on lui fournisse des paramètres. Pour cela, la boîte de dialogue suivante apparaît :

OUVRIR UNE APPLICATION

Nom: **IMPGRAPH.TTP**

Paramètres:

CONFIRMER **ANNULER**

Cela signifie qu'il faut fournir au programme un ou plusieurs paramètres. S'il s'agit d'un éditeur, le paramètre attendu sera son nom et le chemin qui y accède :

Exemple: "A:\TEXTENDOCUMENT.DOC"

Les paramètres ne peuvent pas occuper plus de 64 caractères. L'emploi des majuscules ou minuscules est indifférent sur tous les ST équipés des anciennes ROM. Les nouvelles par contre font la différence. Cela peut éventuellement poser problème si le programmeur n'en n'a pas tenu compte. Dans le doute, employez toujours les majuscules.

○ L'icône de données

Il s'agit d'un fichier ne contenant que des données: texte, nombres, images, ou autre.



Le suffixe employé donne très souvent une idée de son contenu. Enumérons les plus courants et le type de fichier associé (notons que le signe "*" signifie "n'importe quelle suite de caractères devant le point").

SUFFIXE	CONTENU DU FICHIER
*.ASC	Texte pur (suite de caractères).
*.TXT	Texte pur (suite de caractères).
*.DOC	Texte avec paramètres de style et/ou mise en page (produit par un traitement de texte).
*.PAS	Texte source d'un programme en PASCAL.
*.BAS	Texte source d'un programme en BASIC.
*.C	Texte source d'un programme en C.
*.MOD	Texte source d'un programme en MODULA.
*.LIS	Texte source d'un programme en LISP.
*.ASM	Texte source d'un programme en ASSEMBLEUR.
*.PIC	Image quelconque.
*.PII/2/3	Image en basse/moyenne/haute résolution.
*.RSC	Ressources d'un programme GEM (voir plus loin).

Le contenu des fichiers de texte ASCII est directement visualisable en double-cliquant leur icône. Seules les 24 premières lignes apparaissent.

L'appui sur [SPACE] ou [RETURN] provoque respectivement l'affichage d'une ou des 24 lignes suivantes.

La touche "Q" (quit) permet d'arrêter l'affichage et de se retrouver dans le Bureau.

☞ Remarques :

✓ Si le fichier contient autre chose que du texte ASCII (*.DOC, *.PI3), son affichage est toujours possible, mais le résultat sera aberrant.

✓ La plupart des programmes sources, quel que soit le langage, sont stockés en ASCII. Ils sont donc directement visualisables à partir du bureau. Certains interpréteurs Basic (GFA_BASIC) par contre "prédigèrent" le texte source avant de le stocker sous une forme condensée, non-ASCII. Il faudra alors rentrer dans l'éditeur spécifique au langage pour lire ou imprimer le contenu d'un fichier *.BAS.

○ L'icône de Dossier (folder)

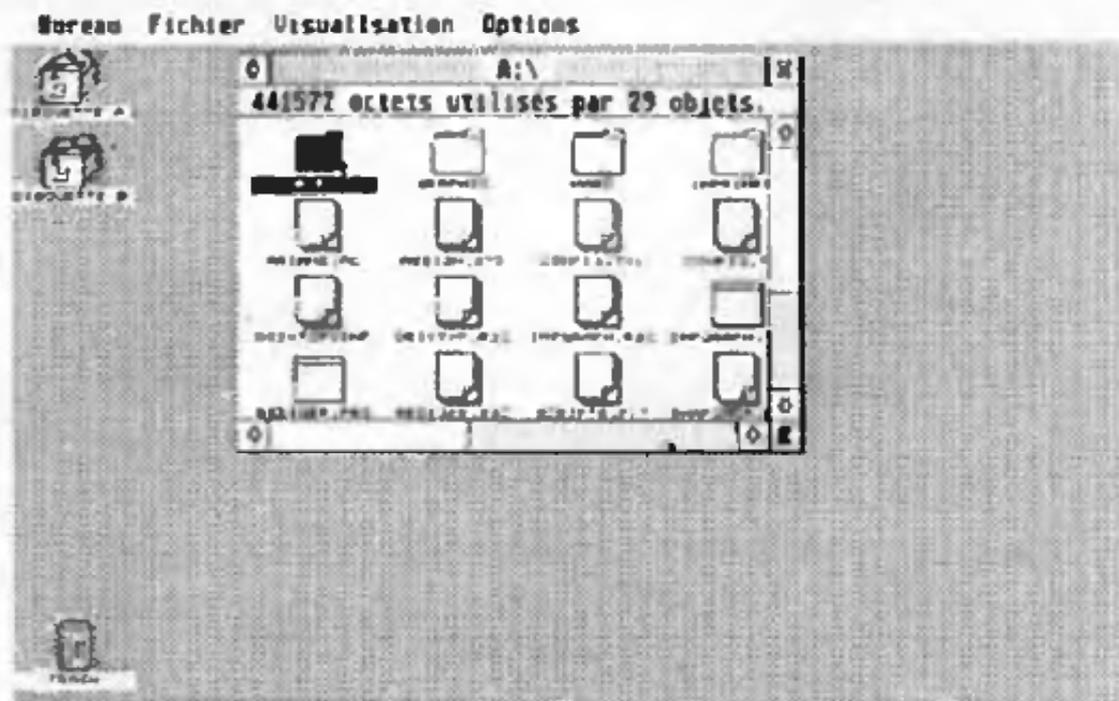
Si l'on compare les fichiers de données et les programmes à des feuilles volantes, alors le dossier est l'analogie de la farde cartonnée.



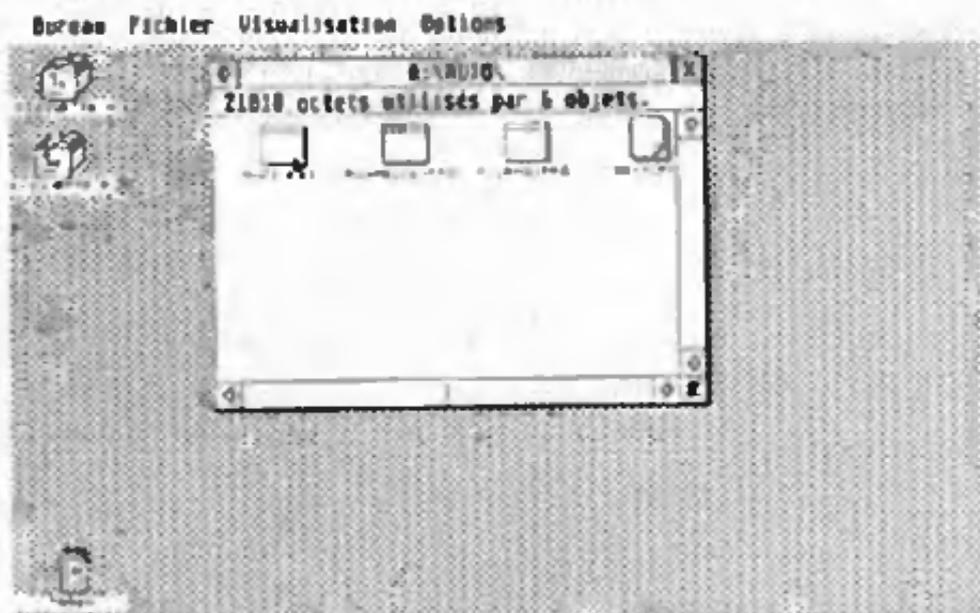
Elle rassemble sous un même titre toutes les feuilles qui se rapportent à un sujet et peut éventuellement contenir d'autres fardes. Cela permet de faciliter le classement et la recherche des fichiers.

⇒ Ouvrir un dossier

On ouvre un dossier d'un double-clic sur son icône :



La fenêtre où il se trouvait est mise à jour et n'affiche plus dès lors que le contenu propre du dossier :



La barre de titre indique maintenant le chemin ("path") parcouru dans notre recherche :

A:\AUTO

Chaque back-slash ("\") indique la descente dans un dossier :

A:

"\" est le dossier de niveau zéro. Il rassemble tout ce que contient la disquette.

AUTO

"AUTO" est un dossier de niveau (profondeur) un.

Remarque :

La ligne d'information indique uniquement la somme des tailles des programmes et fichiers de données visibles au niveau courant. Les dossiers n'étant que des "enveloppes", ils n'ont pas de dimensions. Si

aucun fichier libre - non inclus dans un dossier - n'est présent à un niveau donné, elle indiquera "0 octets utilisés par 0 objets".

➤ Refermer un dossier

Pour remonter d'un niveau dans l'arborescence du catalogue, il suffit d'employer la boîte de fermeture de la fenêtre. On ferme ainsi le dossier qui redevient une simple icône du niveau précédent.

Si l'on quitte le dossier "A:\", cela équivaut à "fermer" la disquette. Cela provoque la disparition - fermeture - de la fenêtre associée au lecteur "A".

➤ Une autre fenêtre

Le bureau vous autorise à avoir jusqu'à quatre fenêtres ouvertes simultanément à l'écran. Consultons le catalogue de la disquette présente dans le deuxième lecteur.

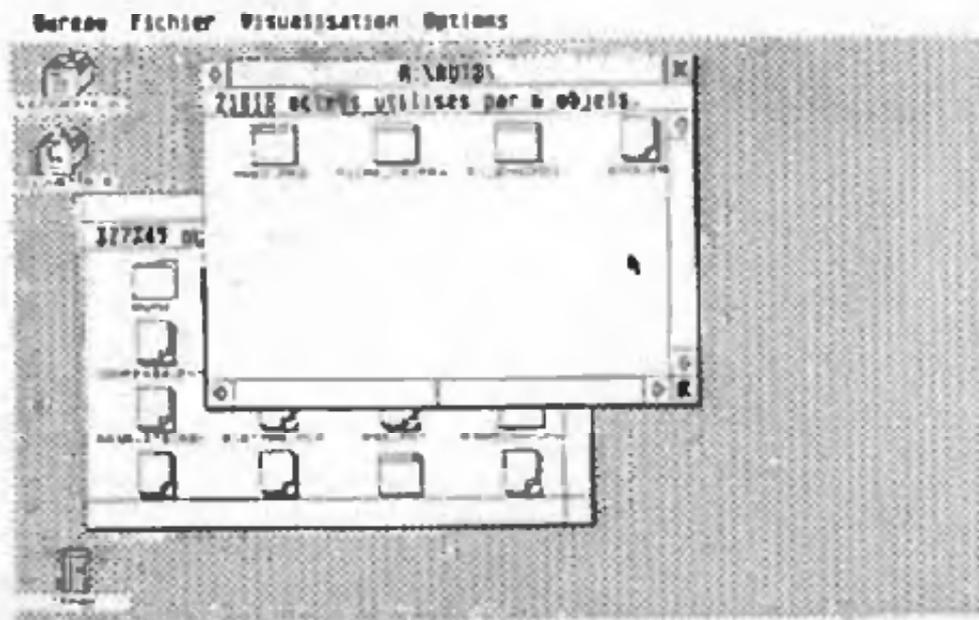
A cette fin, double-cliquons l'icône du lecteur "B":



La barre de titre de la première fenêtre est devenue blanche, ce qui indique qu'elle est inactive. De plus, elle est passée à l'arrière-plan. La fenêtre active - "B" - la recouvre partiellement.

Il n'y a jamais, à un moment donné, qu'une seule fenêtre qui puisse être opérationnelle, ou active. En cas de chevauchement, elle occupe toujours le premier plan.

Pour réactiver une fenêtre présentement inactive, il suffit de la sélectionner (cliquer).



Manipulations de fichier :

⇒ Effacement d'un fichier

Rien de plus simple : mettez-le à la poubelle!



1° Sélectionner le fichier

2° Tirer le fichier vers la poubelle

Pour éviter une erreur malencontreuse, une boîte de dialogue GEM apparaît et vous demande confirmation de l'ordre :

SUPPRIMER DES DOSSIERS/OBJETS

Dossiers à supprimer: ___0
Objets à supprimer: ___1

CONFIRMER

ANNULER

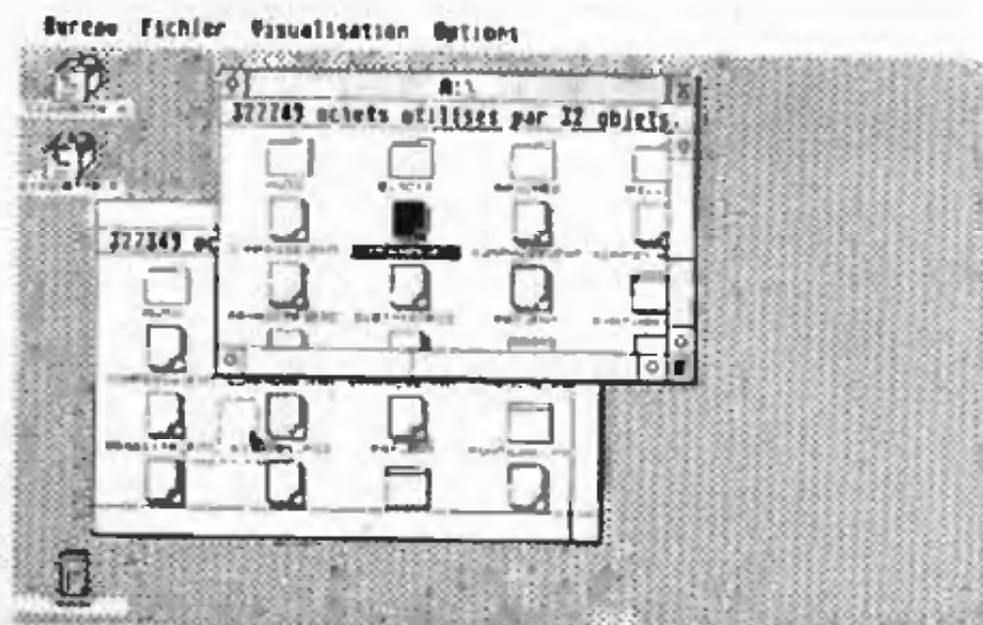
Ⓢ Attention :

La poubelle du bureau est un véritable incinérateur. Une fois détruit, le fichier n'est plus récupérable. Contrairement à MS-DOS et UNIX, le TOS permet de détruire un dossier non-vidé.

⇒ Copie de fichiers

La copie d'un fichier de la fenêtre active vers une autre se fait d'une manière simple :

- 1° Sélectionner l'icône du fichier.
- 2° La tirer dans la fenêtre cible.
- 3° Relâcher la pression.

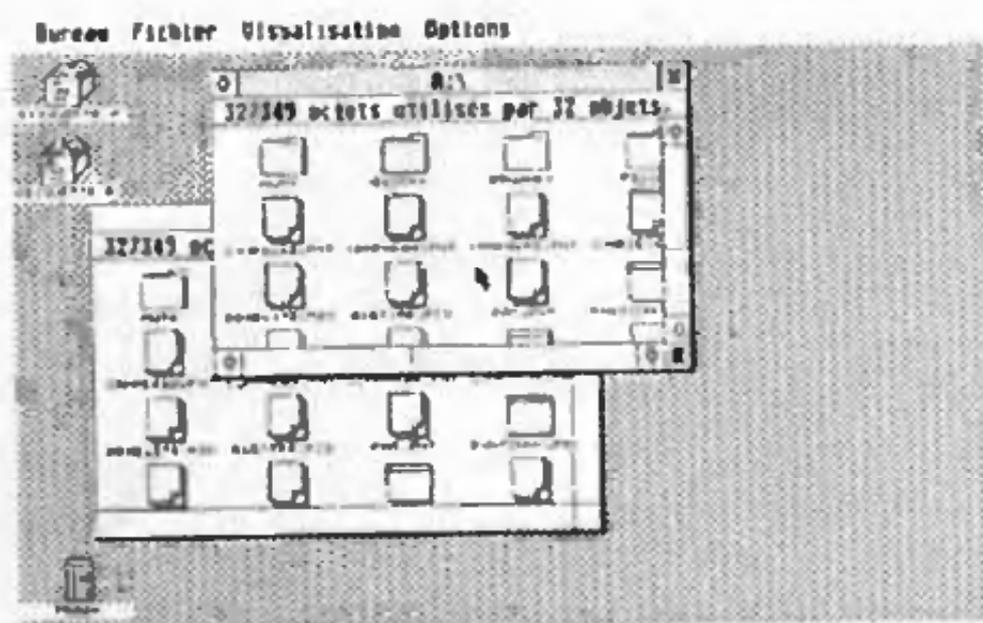


La copie ou la destruction d'un ensemble de fichiers suit la même méthode. Seule la sélection diffère : elle devient multiple.

• Sélection multiple

Pour choisir plusieurs fichiers à la fois, il existe deux méthodes :

1° Les encadrer



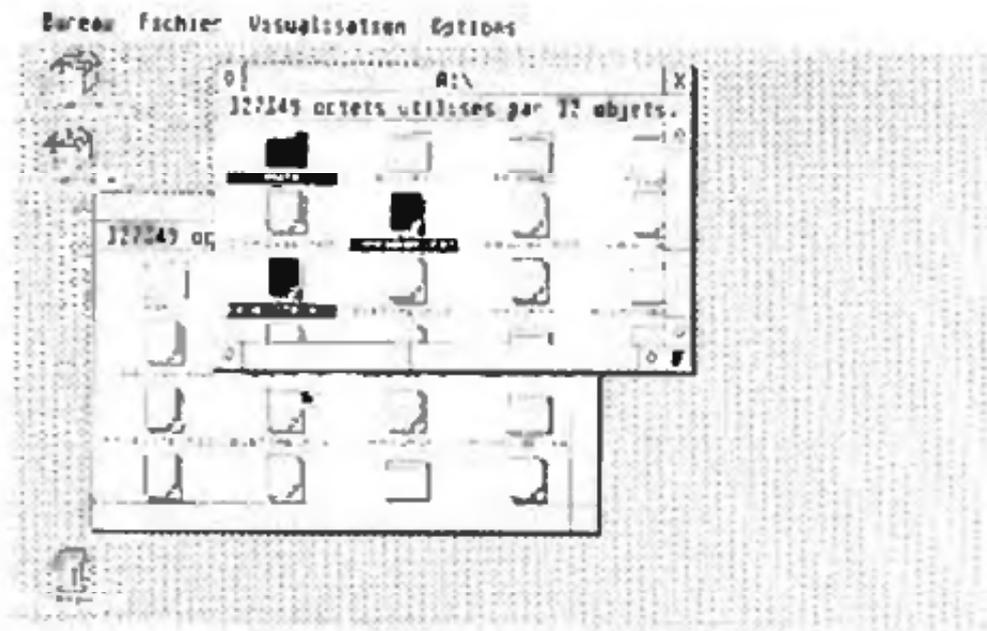
Le cadre qui les englobe est tracé avec la souris de la manière suivante :

- 1° Cliquer en haut à gauche
- 2° Tirer la souris vers le coin inférieur droit
- 3° Relâcher la pression

Toutes les icônes entourées par le cadre noircissent : les fichiers sont sélectionnés.

2° Les "shift-cliquer"

Comme le nom l'indique, il s'agit de cliquer l'icône d'un fichier tout en maintenant la touche [Shift] enfoncée. Il faut répéter cette opération pour chaque fichier.



Remarque

Un *shift-clic* sur un fichier déjà sélectionné le libère.

Le menu Visualisation :

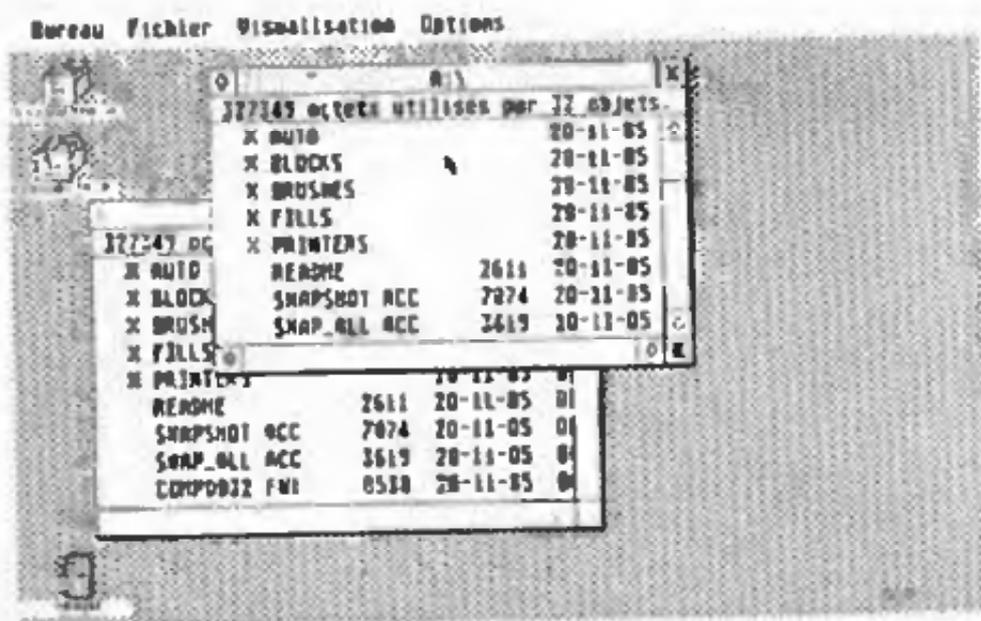
Visualisation

- ✓ Visualisation icône
- Visualisation texte
- ✓ Tri par nom
- Tri par date
- Tri par taille
- Tri par type

En plus de la taille et de la position d'une fenêtre, on peut encore modifier un autre de ses paramètres : la manière dont son contenu est visualisé. C'est le rôle du menu Visualisation. On déroule un menu en

amenant le pointeur sur son titre. On sélectionne une de ses options en la cliquant.

Par défaut, les fichiers sont représentés par des icônes et ordonnés par ordre alphabétique. Lorsque le nombre d'icônes augmente, ce choix s'avère peu pratique. A l'usage, on préférera souvent l'affichage en mode texte, trié sur le type - suffixe - des fichiers. Faisons ces choix. La fenêtre prend l'aspect suivant:



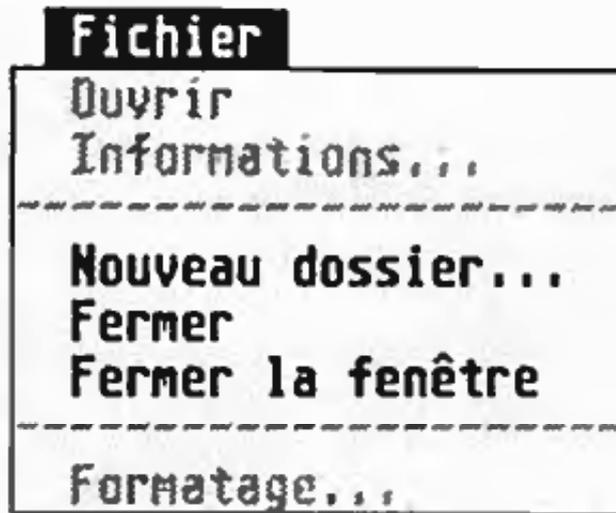
Les noms de dossiers apparaissent en tête de liste et sont précédés d'un caractère graphique. A côté des noms de fichiers apparaissent leurs caractéristiques :

- ✓ Taille en octets.
- ✓ Date et heure de leur dernière modification (écriture).

☞ Remarque :

- ✓ La sélection - simple ou multiple - se fait comme avec les icônes. Il suffit de cliquer au-dessus du nom ou d'une des caractéristiques du fichier.

Le menu Fichier :



Certaines rubriques du menu file apparaissent en grisé. Cela veut dire qu'elles sont inactives pour l'instant. Dans ce cas-ci, c'est parce que les objets auxquels elles s'appliquent n'ont pas été sélectionnés.

⇒ **Ouvrir**

C'est l'équivalent du *double-clic* sur une icône. Pour ouvrir un catalogue par exemple, il faut 1° sélectionner l'icône du lecteur de disquette et 2° cliquer l'option "Open".

⇒ **Informations**

Cette option permet, après avoir sélectionné l'icône *ad hoc* d'obtenir des informations supplémentaires sur une disquette ou un fichier :

Informations d'un fichier (données ou programme): le GEM affiche la boîte suivante :

INFORMATIONS OBJET	
Nom:	SNAP_ALL.ACC
Nombre d'octets:	----3619
Dernière modification:	28/11/85 00:11
Attributs:	Lecture/Ecriture
	Lecture
CONFIRMER	ANNULER

Protéger un fichier :

Un nouvel attribut caractéristique du fichier est visible : la permission de le modifier. Par défaut, le système l'autorise. Si vous voulez empêcher sa modification ou son effacement malencontreux, sélectionnez l'*option [Lecture]*. Elle apparaît alors en vidéo inverse. Si vous essayez par la suite de l'effacer ou de modifier son contenu, la boîte d'alerte suivante apparaît :

	Un objet ayant ce nom existe déjà, ou bien il n'autorise pas l'écriture.
	CONFIRMER 

Pour pouvoir l'effacer, il suffit de réactiver son attribut [Lecture/Ecriture].

Renommer un fichier :

A droite du nom du fichier se trouve le curseur (barre verticale). On efface l'ancien soit caractère par caractère - [Backspace] ou [Delete] -, soit en bloc - [Esc]. Il suffit ensuite d'entrer le nouveau nom.

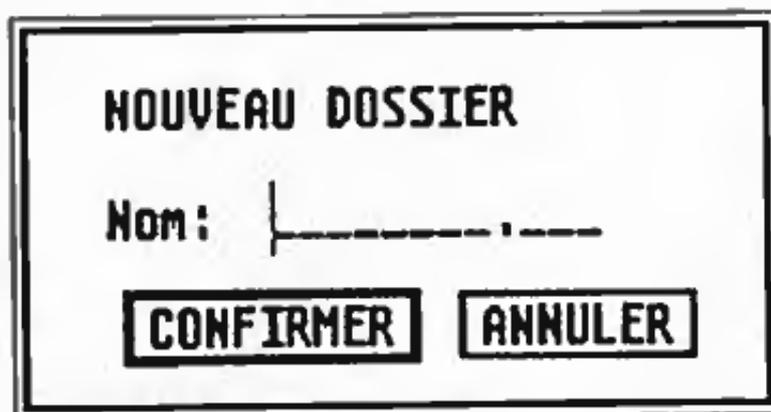
Informations d'un lecteur de disquette :

INFORMATIONS DISQUE	
Nom de l'unité:	A:
Nom du disque:	-----
Nombre de dossiers:	----6
Nombre d'objets:	---74
Octets utilisés:	--359575
Octets disponibles:	--395264
CONFIRMER	

Si vous faites la somme de la place occupée sur la disquette et de la place encore disponible, vous obtenez un nombre inférieur à la capacité maximale de la disquette. C'est dû à la perte de place occasionnée par les *clusters* (voir le chapitre consacré à l'OS : Gestion du disque).

• Nouveau dossier

Cette option crée un nouveau dossier. Elle ouvre la boîte de dialogue suivante :



NOUVEAU DOSSIER

Nom: _____

CONFIRMER **ANNULER**

Il suffit de lui donner un nom et de sélectionner la case [CONFIRMER]. Si vous choisissez, la case [Cancel], l'opération est annulée.

• Remarque :

La case [OK] est tracée en traits épais. Cela signifie que l'appui de la touche [Return] produit le même résultat que sa sélection avec la souris.

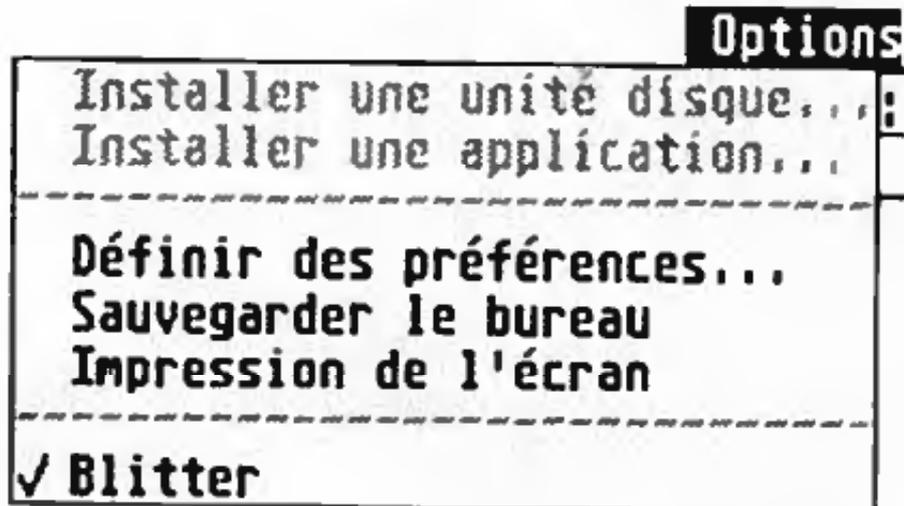
• Fermer

C'est l'équivalent de la case de fermeture de la fenêtre. Elle vous fait remonter d'un niveau dans la hiérarchie des fichiers et ferme la fenêtre si c'est le dernier.

• Fermer fenêtre

Cette option ferme la fenêtre.

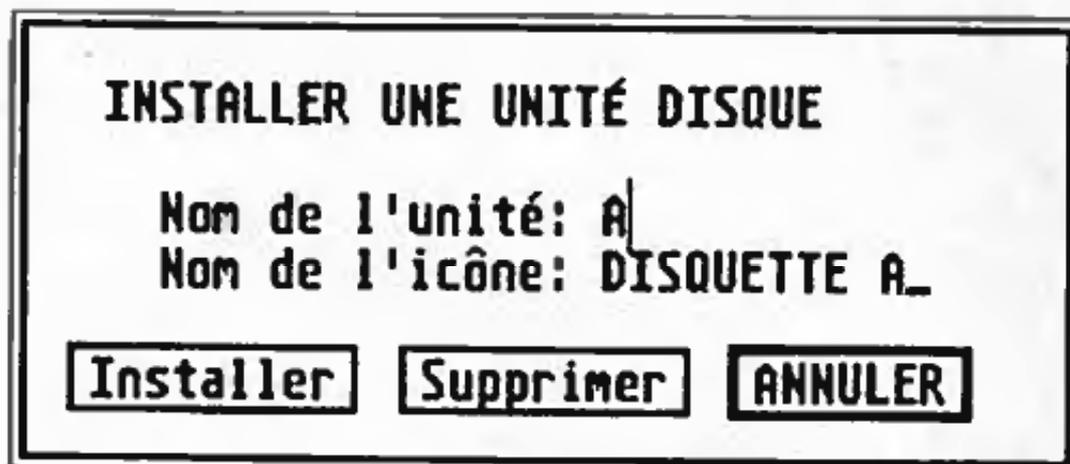
Le menu Options :



Il vous est permis de modifier certains paramètres par défaut et/ou l'aspect du Bureau.

❖ **Installer une unité disque**

Crée, supprime ou renomme une icône de lecteur :



○ Supprimer une icône :

Fait disparaître une icône de lecteur du bureau :

1. Sélectionner l'icône à supprimer.
2. Cliquer [Supprimer].

○ Créer une icône :

Fait apparaître une icône de lecteur.

1. Sélectionner une icône.
2. Remplacer l'ancien identificateur par celui de la nouvelle icône.
3. Lui donner un nom (facultatif).
4. Cliquer [Installer].

L'identificateur est une lettre majuscule comprise entre A et P.

A, B : lecteurs de disquettes.

C...P : partitions de disque dur.

☞ Remarque:

Vous pouvez indifféremment taper l'identificateur en majuscule ou minuscule, sauf si c'est un "c". En effet, "C" désigne le disque dur, et "c" le port cartouche.

⇒ Renommer une icône

Procéder comme ci-dessus, mais ne pas modifier l'identificateur.

☞ Remarque:

Si vous supprimez toutes les icônes de votre bureau, vous n'avez plus aucun accès aux lecteurs. Il faut alors réinitialiser le système.

⇒ Installer une application

S'il est une fonction du bureau méconnue, c'est bien celle-ci. Il est vrai qu'elle sort un peu de l'ordinaire.

○ Rôle :

[Installer une Application] vous permet de fusionner deux opérations très fréquentes en une. Prenons un exemple concret.

Le traitement de texte *Le Rédacteur* sauve les textes créés dans des fichiers de type "DOC". Chaque session de travail avec lui commence invariablement de la même manière :

- 1° Vous double-cliquez l'icône du programme.
- 2° Le programme se charge en mémoire.
- 3° Le programme vous demande le nom du fichier à modifier.
- 4° Vous sélectionnez le nom du fichier.
- 5° Le programme le charge en mémoire.
- 6° Vous pouvez commencer à travailler.

La fonction [Installer une Application] permet d'accélérer cette manoeuvre et la remplace par :

- 1° Vous double-cliquez l'icône du fichier.
- 2° Le programme se charge en mémoire.
- 3° Le fichier est chargé en mémoire.
- 4° Vous pouvez commencer à travailler.

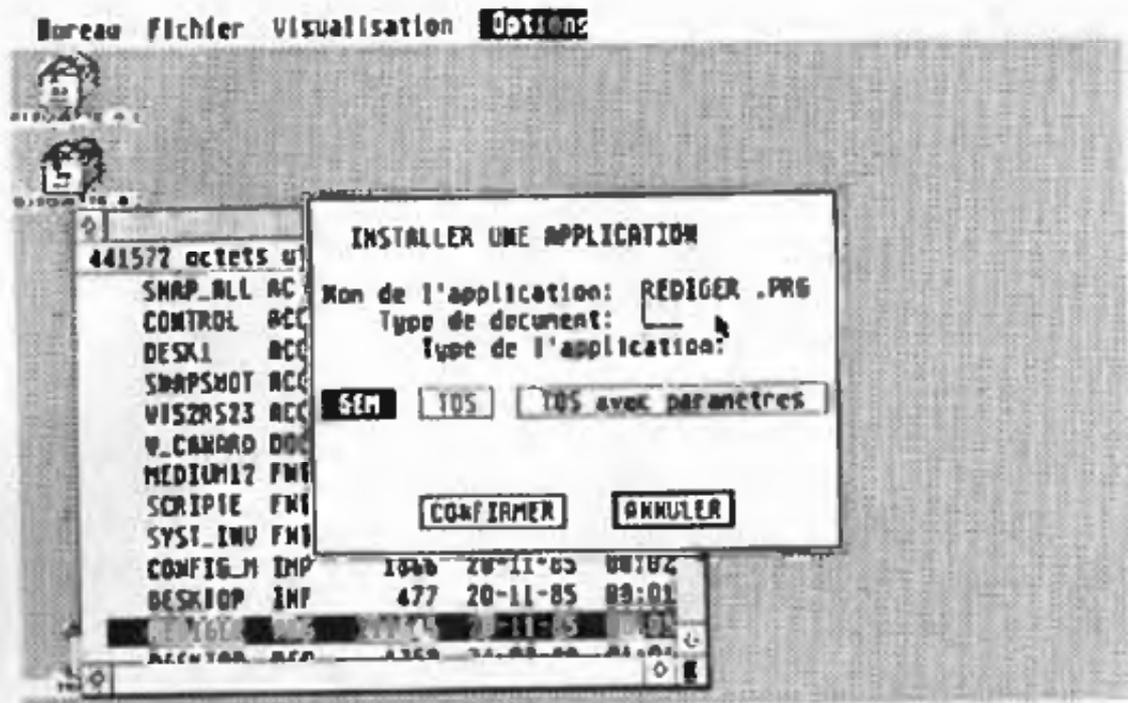
Il suffit pour cela d'associer tous les fichiers de type "DOC" au programme qui les manipule.

☛ Comment "Installer" un logiciel ?

La manoeuvre est simple

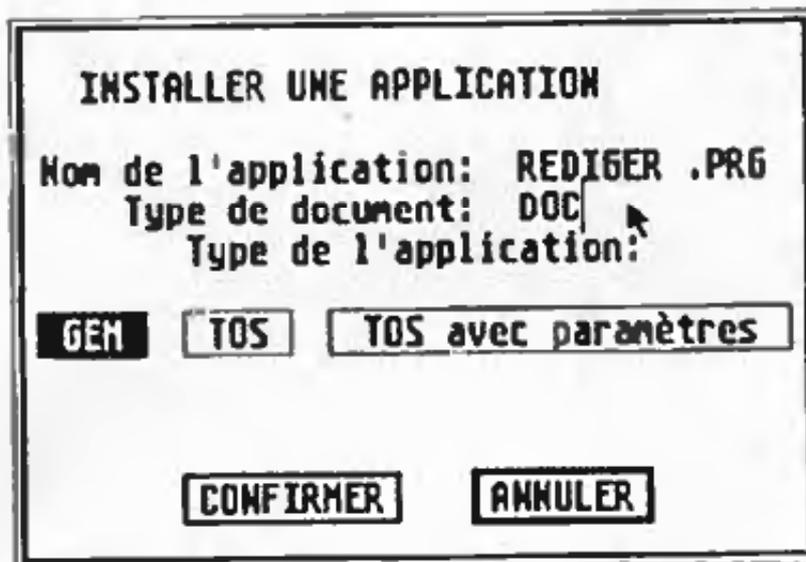
- 1° Sélectionnez le programme.
- 2° Sélectionnez [Install Applications].

Une boîte de dialogue apparaît :



3° Entrez le type - suffixe - (DOC dans notre exemple) des fichiers à lui associer.

4° Entrez le type du programme.



5° Cliquez [OK].

Dorénavant, chaque fois que vous double-cliquerez l'icône d'un fichier de type "DOC", GEM saura qu'il doit enchaîner 1° le chargement du programme et 2° le chargement du fichier.

☞ **Remarques :**

✓ On ne peut installer qu'un seul programme.

✓ On peut employer des "Jokers" - wildcards - dans le type de fichier.

"DO?" associe au programme tous les fichiers dont le suffixe est "DO" + un caractère quelconque.

"D*" associe au programme tous les fichiers dont le suffixe débute par "D".

✓ Pour conserver l'installation d'une application d'une session de travail à une autre, il faut sauver les paramètres du bureau (voir plus loin).

✓ Tout programme n'est pas forcément "instalable". Si son manuel d'utilisation n'en parle pas, c'est qu'il ne supporte pas cette facilité.

✓ **Le type de l'application :**

GEM : tout programme de type "PRG" qui dispose d'une barre de menu (et utilise donc le GEM).

Tos Takes Parameters : tout programme de type ".TTP" ou ".APP".

TOS : tous les autres.

➔ Définir des préférences



- ➔ **[Confirmer la Suppression]:** Faut-il demander confirmation lors des copies de fichiers ?
- ➔ **[Confirmer les Copies]:** Faut-il demander confirmation lors des effacements de fichiers ?
- ➔ **[Résolution écran]:** Si vous possédez un moniteur couleur, cette option vous permet d'installer une autre résolution : base ou moyenne.

➔ Sauvegarder le bureau

En personnalisant notre bureau, nous avons changé son aspect et ses paramètres par défaut. Pour retrouver ces choix lors des prochaines sessions de travail, il faut les sauver sur la disquette de démarrage. C'est ce que réalise [Save Options]. La "disquette de boot" (de démarrage) est celle que l'on place dans le lecteur "A" avant d'allumer l'ordinateur (voir plus loin : le boot).

Les différents paramètres sont sauvés dans le fichier **DESKTOP.INF** qui est sur la disquette de démarrage.

➔ Impression de l'écran

Si vous avez connecté une imprimante matricielle ou laser, cette option provoque l'impression de l'image

d'écran. Le même résultat peut être obtenu en pressant simultanément les deux touches [Alternate][Help]. Une seconde pression sur ces deux touches stoppe l'impression en cours.

⇒ **Blitter**

Cette option n'existe que sur les MEGA-ST. Elle permet de désactiver le blitter. Ceci permet d'exécuter certains "vieux" programmes qui ne supportent pas la présence du Blitter.

Le menu Bureau

Jusqu'à présent, les menus que nous avons rencontrés nous permettaient de manipuler des fichiers ou de paramétrer le Bureau. Le menu Desk est, lui, tout à fait différent. Les options qu'il propose n'ont rien à voir avec le bureau et l'OS. Ce sont des programmes d'un genre spécial que l'utilisateur peut employer à sa guise. Ils sont chargés lors du chargement (boot) du système.

○ Le chargement (boot)

Lorsqu'on allume l'ordinateur, une suite d'opérations s'enchaînent automatiquement : le bootstrapping, ou boot (amorçage, initialisation).

1° TOS.IMG

Tout d'abord, le système regarde si la disquette de boot (amorçage) contient le fichier "TOS.IMG" : C'est le système d'exploitation. Si oui, il est chargé en mémoire. Cette étape a lieu même si vous possédez l'OS en ROM.

2° AUTO

Les programmes de type "PRG" présents dans le dossier "AUTO" sont exécutés. L'ordre suivi est celui dans lequel ils ont été copiés dans le dossier. Certains programmes - Turbodos, par exemple - doivent impérativement être exécutés avant tous les autres. Ils faut donc les copier en premier dans le fichier AUTO. Les programmes de type "TOS", "TTP" et "APP" ne sont pas pris en compte.

Pendant cette étape, GEM n'est pas encore disponible. C'est pour cela que les programmes en question n'utilisent ni fenêtre, ni menu et même pas de souris.

3° ACCESSOIRES

Les six premiers accessoires présents sur la disquette de boot sont chargés en mémoire et exécutés.

Il arrive parfois que deux accessoires ne se supportent pas. La procédure de boot se met à "cycler" - recommencer indéfiniment. Il faut alors éteindre l'ordinateur, booter avec une autre disquette et supprimer un des accessoires litigieux.

4° LE BUREAU

Si le fichier "DESKTOP.INF" est présent sur la disquette de boot, il est chargé. On retrouve alors toutes les options que l'on avait sauvegardées.

⇒ Le Reset

Une petite proéminence orne le dos de l'ATARI ST : c'est le bouton de RESET. Il permet de "re-booter" sans toucher à l'interrupteur.

⇒ **Activer un accessoire**

Il s'agit d'un programme de type "ACC" qui est chargé lors du boot et qui reste en mémoire jusqu'à l'extinction de la machine. Il est normalement inactif. On le réveille en le sélectionnant dans le menu Desk. L'application en cours s'arrête et l'accessoire s'exécute. Quand il a fini son travail, il disparaît et l'application est réactivée.

⇒ **Supprimer un accessoire**

Si vous n'utilisez jamais un accessoire, il ne sert à rien de le charger. Il occupe inutilement de la place et son chargement allonge la durée du boot. Il y a deux manières d'empêcher son chargement : l'effacer ou changer son suffixe (voir renommer un fichier). Il est pratique de renommer le "ACC" en "AC". Si on désire par la suite réemployer cet accessoire, il suffit de changer le "AC" en "ACC".

⇒ **Les accessoires standards**

La disquette de boot livrée avec l'ATARI ST contient plusieurs accessoires standard. Ils permettent de modifier certains paramètres supplémentaires du système tels que l'heure, la date, le type d'imprimante utilisée, etc. C'est ce que nous allons voir maintenant.

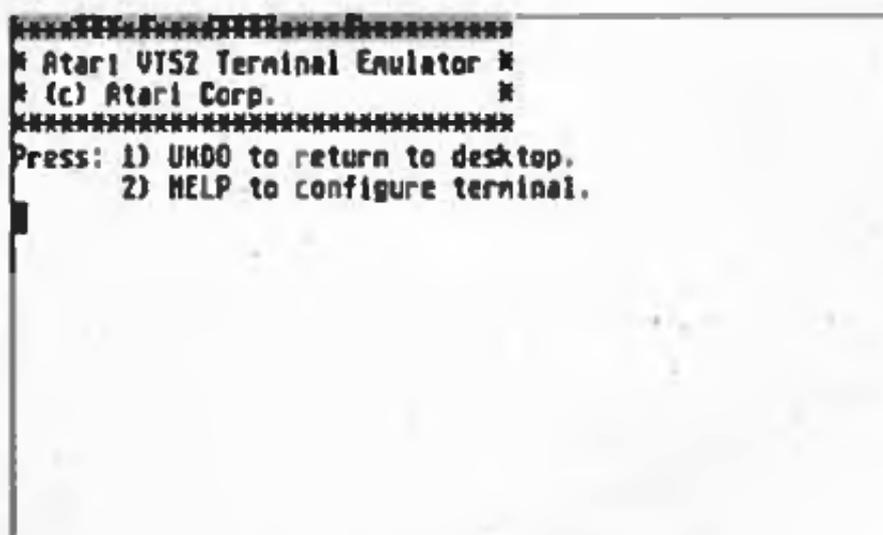
➔ Informations bureau



C'est le copyright et le nom de l'application en cours qui est, dans ce cas-ci le bureau. Lorsque vous chargez un autre programme GEM, il place ici son propre message.

➔ Emulateur VT-52

C'est un programme de communication qui vous permet d'envoyer et de recevoir des données via le port-modem (RS-232). Lorsque vous sélectionnez cette option, l'écran suivant apparaît :



A tout moment, trois actions sont possibles :

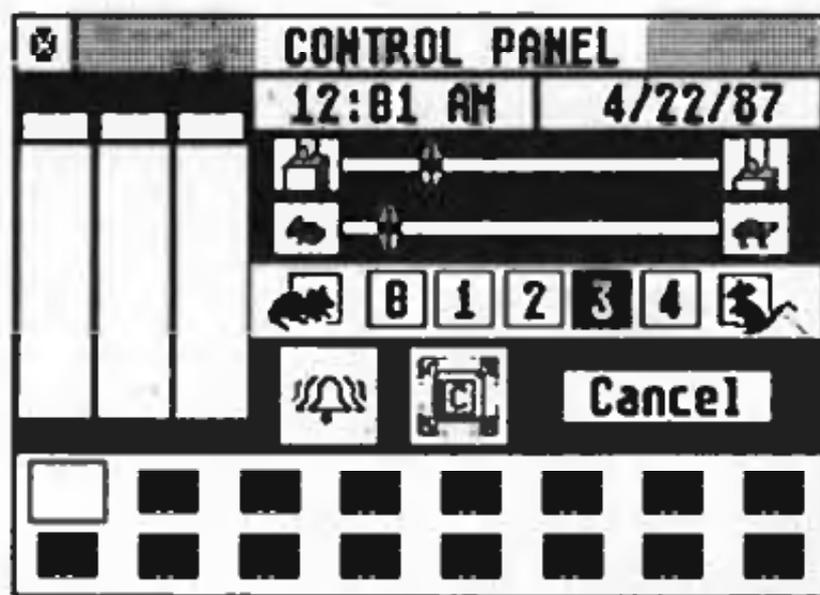
- **Communiquer** : ce que vous tapez au clavier est envoyé à travers le port RS-232, et ce qui en vient est affiché sur votre écran.

○ **Paramétrer** le port RS-232 (voir Set RS-232 Config).

○ **Quitter l'émulateur** : vous revenez dans le bureau.

Ce programme de communication est élémentaire mais il vous permet, sans frais aucun, d'employer directement votre ST à la place d'un terminal VT-52.

↳ Control panel



Il permet de modifier 5 paramètres importants du système :

○ **Heure et Date** :

Le 520 et 1040 n'ont pas d'horloge permanente. Il faut dès lors les mettre à l'heure au début de chaque session.

○ **Répétition automatique du clavier** :

Toutes les touches du clavier sont à répétition automatique. Vous pouvez modifier le délai avant la répétition (ligne supérieure) et la vitesse de répétition.

○ **Vitesse du double-clic** :

Un double-clic est une succession rapide de deux clics. Si vous n'allez pas assez vite, l'ordinateur l'interprète

alors comme deux clics séparés. Vous pouvez lui demander d'allonger (vers la droite) ou de raccourcir (vers la gauche) le délai entre les deux pressions de boutons.

○ Signaux auditifs :

Ils sont deux : le "clic", accompagnant chaque pression de touche et la "clochette" après une action erronée au clavier ou avec la souris. Les deux sont désactivables séparément. Si l'icône apparaît en affichage léger, c'est que l'option est inactive.

○ Palette de couleurs :

S'applique seulement au moniteur couleur. Il permet de modifier les couleurs d'affichage en faisant varier la proportion des trois couleurs primaires (rouge, vert, bleu). Ce point sera détaillé dans le chapitre consacré au *GRAPHISME*.

○ Cancel :

Redonne aux cinq paramètres les valeurs qu'ils avaient avant modifications.

On quitte le panneau de contrôle en cliquant la case de fermeture de la fenêtre.

☞ Remarque:

En examinant la fenêtre du panneau de contrôle, vous remarquerez qu'elle est plus simple que celles rencontrées jusqu'ici : une seule case - de fermeture -, pas de ligne d'information, ni d'ascenseur. En réalité, il n'y a que la barre de titre qui soit obligatoire. Toutes les autres options sont facultatives.

⇒ RS-232

RS232 PORT CONFIGURATION	
Baud Rate:	300 400 1200 2400
Parity:	None Odd Even
Duplex:	Full Half
Bits/Char:	8 7 6 5
Strip Bit:	On Off
Flow Control	
Xon/Xoff:	On Off
Rts/Cts:	On Off
OK	Cancel

Aussi appelé *port série* ou *port modem*, le port RS-232 permet d'échanger des données avec le monde extérieur. Cet accessoire vous permet de modifier certains paramètres de la communication. Les plus importants sont la vitesse - *Baud Rate* - en bits par seconde, la *parité* et la taille des caractères, en bits. L'option *Full Duplex* signifie que l'on peut émettre et recevoir des données en même temps (comme au téléphone). En *Half Duplex*, chacun "parle" à son tour (comme en CB).

XON/XOFF et RTS/CTS sont des protocoles qui permettent de rattraper certaines pertes d'information causées par une mauvaise qualité de la ligne téléphonique.

⇒ Install printer

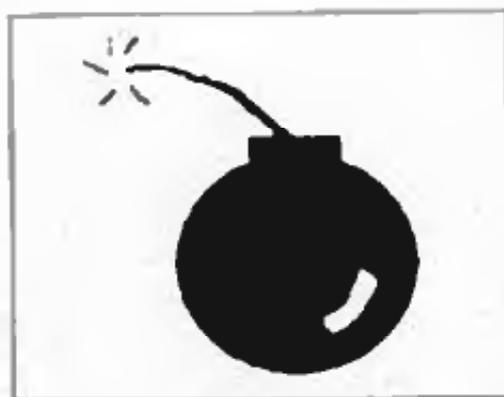
INSTALL PRINTER	
Printer Type:	Dot DotM
Color:	B/W Color
Pixels/Line:	1200 900
Quality:	Craft Fine
Printer Port:	Printer Modem
Paper Type:	1000 Single
OK	Cancel

Par défaut, l'ordinateur suppose que vous employez une imprimante matricielle (*Dot matrix*), monochrome (*B/W*), que son alimentation en papier est continue (*Feed*) et qu'elle est branchée sur le port Centronics (*Printer*). Les caractères sont imprimés en qualité normale (*Draft*). En mode graphique, une ligne horizontale contient 1280 points.

Vous pouvez changer ces options et utiliser une imprimante à marguerite (*Daisy*) branchée sur le port série (*Modem*), l'alimenter feuille par feuille (*Single*) et imprimer les caractères en qualité courrier (*Final*). Notons pour terminer que l'imprimante couleur n'est pas supportée par le système. Elle nécessite un *driver* (logiciel de communication), personnalisé. Les imprimantes sont examinées en détails dans le chapitre consacré aux *PÉRIPHÉRIQUES*.

Bombes, fenêtres et icônes

☛ Les bombes



On n'a jamais vu ça, un ordinateur qui lance des bombes! Oh, rassurez-vous, rien de bien dangereux, si ce n'est pour vos données.

Si un de vos programmes commet une mauvaise action, l'ordinateur réagit : il stoppe son exécution et le TOS affiche à l'écran une rangée de bombes.

Une "mauvaise action" est une manœuvre illégale. Il est interdit, par exemple de diviser un nombre par zéro. Si vous essayez, vous recevez cinq bombes à l'écran. Au mieux, votre programme s'arrête et vous retournez dans le bureau. Au pire, l'ordinateur se bloque. La seule solution, dans ce cas, est de réinitialiser le système (Reboot).

☛ Les fenêtres

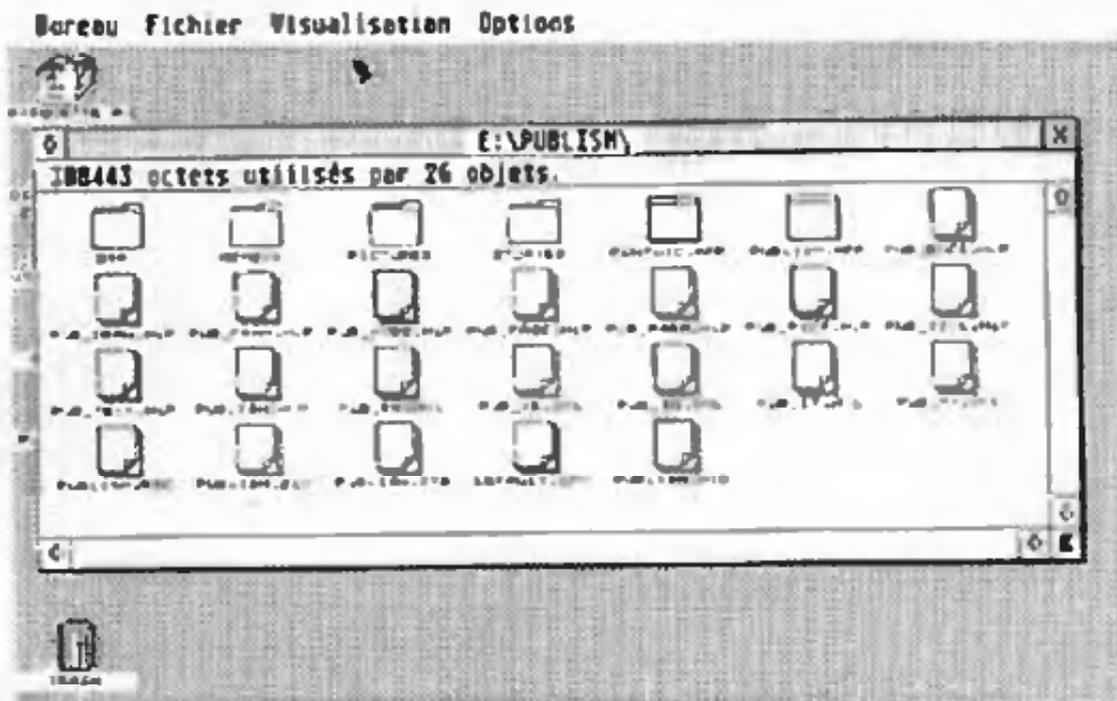
Le bureau vous permet d'avoir jusqu'à quatre fenêtres présentes simultanément à l'écran.

Cette limite n'est pas imposée par le GEM, mais par le bureau lui-même. En réalité, GEM permet d'ouvrir jusqu'à sept fenêtres en même temps. Certaines applications en font usage.

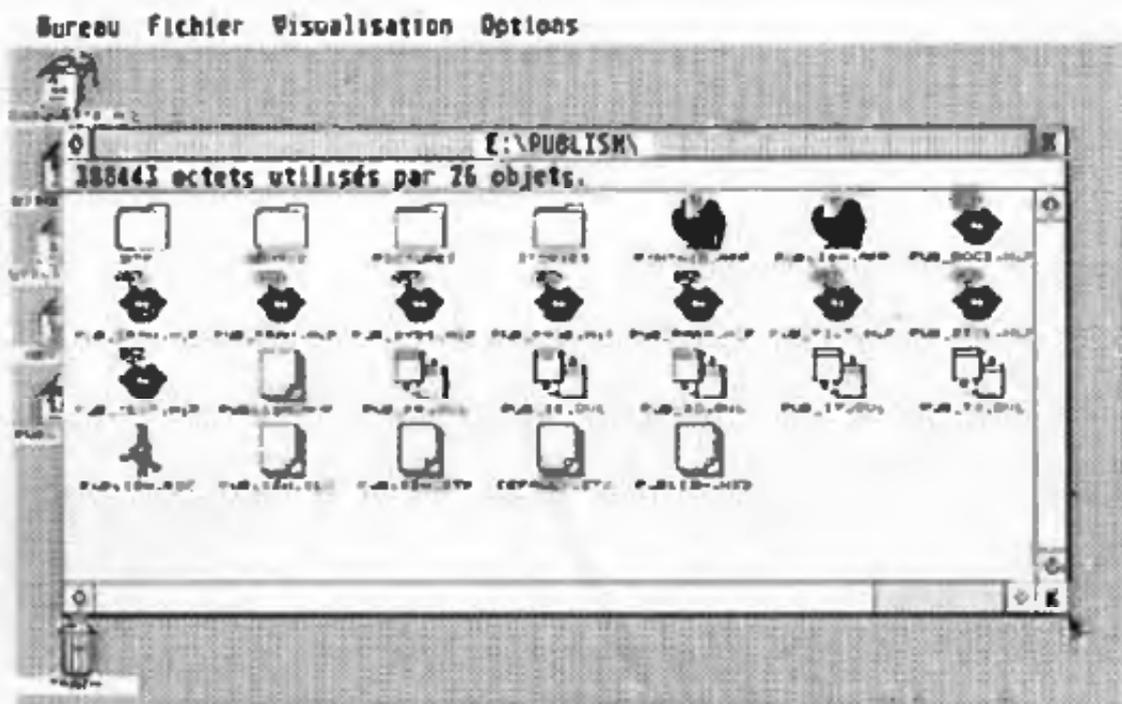
☛ Les icônes

A l'origine, GEM ne connaît que cinq types d'icônes : lecteur, poubelle, dossier, programme et fichier quelconque. Lorsqu'une fenêtre en mode icône est encombrée, le repérage d'un fichier particulier peut s'avérer laborieux. Il existe heureusement un utilitaire du domaine public qui permet de pallier ce manque de diversité : **MAKEIKON.PRG**

➤ Avant modification:



➤ Après modification :



Le programme MAKEIKON permet d'associer aux fichiers des icônes personnalisées. Une icône peut être associée à :

- ✓ un fichier particulier (exemple : GRAPH_01.PRG);
- ✓ un ensemble de fichiers (exemple : *.PRG).

L'espace supplémentaire occupé et le ralentissement d'affichage étant négligeables, on peut se demander pourquoi ce programme ne fait pas partie du système.

2. GEM pour le programmeur

Le bureau est un programme (presque) comme un autre qui se sert des facilités offertes par GEM.

$$\Rightarrow \text{GEM} = \text{AES} + \text{VDI}$$

Le GEM est un ensemble de fonctions mises à la disposition du programmeur. Ces facilités sont réparties entre le VDI et l'AES. Détaillons-les.

↳ VDI

La *Virtual Device Interface* (interface pour périphériques virtuels) constitue la partie "G" - graphique - du GEM.

La VDI est constituée d'un ensemble de fonctions servant à dessiner sur un périphérique virtuel. Avant d'introduire cette notion de périphérique virtuel et ses avantages, examinons quelques-unes des fonctions.

⊃ Fonctions de la VDI

Citons les principales pour donner une idée de la puissance du système mise à la disposition du programmeur.

Ces fonctions permettent entre autre de :

○ *Tracer des formes graphiques :*

Points, lignes brisées, rectangles, ellipses, arcs de cercle, camemberts, etc. On peut, pour chaque forme,

faire varier l'épaisseur et la couleur des traits. Les lignes peuvent être continues ou pointillées, de couleur et d'épaisseur quelconques. L'intérieur des figures fermées (cercle, rectangle,...) peut être rempli de vide (transparent), de couleur ou d'un motif.

Le dernier motif est un motif personnel que l'utilisateur peut créer/modifier à son gré.

○ *Ecrire du texte :*

dans une ou plusieurs polices (*fonts*) de caractères simultanément sur un même écran. Pour chacune d'elles, on peut faire varier la taille des caractères, l'orientation de la ligne d'écriture, l'espacement entre les lignes, les caractères, les mots et bien d'autres paramètres encore. On peut de plus pour chaque caractère combiner les effets spéciaux suivants : épaisseur (gras ou normal), léger (foncé ou clair), souligné, contourné, ombré, penché (italique).

○ *Manipuler des zones d'écran :*

Comme nous le verrons dans le chapitre *GRAPHISME*, l'écran correspond à un bloc de 32 Ko de la mémoire. Recopier une zone rectangulaire de l'écran vers un autre endroit se ramène donc à la copie d'un ensemble de bits de la RAM vers un autre endroit de la RAM. C'est une fonction d'un usage fréquent : le scrolling de texte, par exemple, consiste à décaler (recopier) toutes les lignes sauf une vers le haut ou le bas et à en afficher une nouvelle. Cette opération n'est pas très rapide, ce qui explique la lenteur relative du scrolling des programmes GEM. Une manière de l'accélérer est d'employer le *coprocesseur graphique* (voir le chapitre consacré au *HARD* : Coprocesseur).

▷ Périphérique virtuel

Un périphérique virtuel est une abstraction de la notion de périphérique graphique virtuel (écran, imprimante). On considère uniquement l'idée de périphérique de sortie graphique, sans se soucier de connaître son type (écran, imprimante, traceur, caméra, fichier) ou ses caractéristiques (couleur/monochrome, résolution). Cette abstraction permet d'employer toutes les fonctions indistinctement sur tous les périphériques. Le programmeur n'a pas à se soucier des détails. Ceux-ci sont pris en charge par des drivers.

Un driver est la "concrétisation" du périphérique virtuel. C'est un programme qui se charge de piloter un type de périphérique bien précis: imprimante, écran, etc. A l'origine, le GEM possède des drivers pour la souris, le joystick, le clavier, l'écran, le lecteur de disquettes et l'imprimante, le RS-232 et l'interface MIDI. Si l'on en veut d'autres, il faut les charger en mémoire.

☛ AES

A l'origine, le but des concepteurs du GEM était de développer une interface graphique homme-machine : c'est le *bureau*. Ils ne se sont pas arrêtés là. Ils ont mis à la disposition du programmeur tous les outils - les *routines* - qu'ils avaient créés pour le bureau. En fonction du type de services qu'elles rendent, ces routines ont été regroupées en onze *bibliothèques*.

En plus de ces routines, l'*AES - Application Environment Services* - dispose de routines lui permettant de faire une forme limitée de multitâche: la gestion des accessoires.

Examinons quelques-unes des bibliothèques et leurs principales fonctions :

☛ Window

Création, déplacement et redimensionnement des fenêtres.

☛ Menu

Affichage de la barre de menu, défilement d'un menu, sélection d'une rubrique, etc.

☛ Form

Affichage des boîtes de dialogue et d'alerte.

☛ Graphics

Dessin des ombres rectangulaires qui accompagnent les déplacements ou redimensionnements de fenêtres et d'icônes.

☛ Event

Un événement GEM est un des suivant :

- ✓ Pression d'une touche au clavier.
- ✓ Mouvement de la souris.
- ✓ Pression d'un bouton de la souris.

✓ Ecoulement d'un délai.

Les routines d'Event permettent d'attendre un de ces événements.

Les autres bibliothèques mettent en oeuvre des principes beaucoup plus techniques.

3. L'ATARI sans GEM

Qui peut le plus, peut le moins :

Il est possible de se passer du GEM, que ce soit pour les accès à l'OS ou pour les programmes. On se retrouve alors devant un écran classique, blanc, sans fenêtre ni menu ni icône.

⇒ Pourquoi se passer du GEM ?

Premièrement par facilité: développer une application sous GEM augmente grandement sa convivialité mais demande plus de travail au programmeur.

Deuxièmement pour la portabilité: si l'on veut adapter rapidement sur ATARI ST un programme venant d'un autre ordinateur, sans GEM, le plus simple est de s'en passer. Il faudrait sinon réécrire tout ce qui concerne les entrées/sorties avec la console.

Regardons ce que signifie pour l'utilisateur "se passer du GEM":

O Les programmes non-GEM :

Il existe sur ST quelques programmes qui n'utilisent pas le GEM. Ils se justifient essentiellement dans quatre cas :

1° L'adaptation stricte au ST de programmes non-GEM venant d'un autre ordinateur.

2° Les programmes conçus sur ST devant être portés sur une autre machine.

3° Pour économiser de la place.

4° Les programmes à placer dans le fichier AUTO ne peuvent pas utiliser GEM.

O Accéder à l'OS :

Sans le GEM, l'accès à l'OS se fait par l'intermédiaire d'un interpréteur de lignes de commandes (C.L.I. Command Line Interpreter). Ce programme remplace le GEM et la souris. Les commandes à l'OS (copie de fichier, formatage de disquette, etc.) sont alors passées uniquement au moyen du clavier, ce qui est moins confortable.

L'ATARI ST n'est pas doté d'origine d'un CLI. On en trouve une multitude dans les programmes de domaine public, qui émulent les interfaces MS-DOS ou UNIX.

✓ Désavantages :

Une perte de confort évident, l'obligation de mémoriser la syntaxe exacte de toutes les commandes et de les entrer au clavier sans fautes de frappes.

✓ Avantages :

Le Bureau est une interface figée. Il se veut simple, instinctif, général, mais il ne peut pas s'adapter aux besoins particuliers de l'utilisateur. Les CLI offrent aux "spécialistes" plus de souplesse. Il est possible de les personnaliser, de regrouper des séquences de commandes "batch", de créer des automatismes. Ce genre de facilités se révèle très pratique pour certaines applications qui manipulent beaucoup de fichiers différents et/ou qui exécutent souvent les mêmes séquences de programmes. Les sessions de programmations C en sont un exemple.

☛ L'environnement de travail

☛ Le SHELL

Les compilateurs sont souvent accompagnés d'un "SHELL" ou *environnement de travail*. Il s'agit d'un programme GEM, comme le bureau, qui dirige la session de travail. Il retient les paramètres par défaut - localisation du compilateur, debugger, éditeur -, les noms des fichiers utilisés et permet d'enchaîner facilement les différentes étapes. Edition, compilation, lien des modules et exécution du programme compilé se font à partir d'une unique barre de menu - celle du SHELL - facilement et automatiquement.

☛ Le MAKE

Les environnements de travail offrent aussi souvent un "MAKE". C'est une facilité analogue à la commande homonyme d'Unix. Lorsqu'on crée un programme de taille importante, il est d'usage de le scinder en plusieurs modules plus petits, qui peuvent à leur tour être divisés en plusieurs sous-modules. Une arborescence plus ou moins complexe est ainsi créée. Chaque module est compilé séparément et ils sont tous réunis ("linkés") par la suite pour créer le programme final. Si un de ces modules est modifié, il faut le recompiler ainsi que tous ses ascendants. C'est ce que fait MAKE : il repère les fichiers modifiés et lance automatiquement toutes les recompilations nécessaires, avant de les lier.

Ceci termine le chapitre consacré au GEM.



LES LANGAGES
de
PROGRAMMATION



I. NOTIONS GÉNÉRALES :

Avant d'aborder chaque cas individuellement, rappelons certaines notions générales communes à tous les langages. Ces notions concernent les sujets ci-après:

- ✓ programme;
- ✓ langages et génération;
- ✓ interpréteur/compilateur;
- ✓ éditeur;
- ✓ debugger;
- ✓ ressources;
- ✓ mélange de langages.

Par la même occasion, nous répondrons brièvement à une question d'intérêt général:

- ✓ Pourquoi différents langages ?

☛ PROGRAMME

A de rares exceptions près, un programme est conçu dans le but d'être traité par un ordinateur. A partir de données, il exécute une séquence d'actions qui respectent les directives du programme et produisent le résultat.

☛ LANGAGES ET GÉNÉRATIONS

Un programme est tout d'abord un texte. Comme il décrit un processus plus ou moins complexe, la signification de ce texte doit en respecter les moindres

détails. Cette précision exige un formalisme rigoureux, décrivant les règles à respecter dans l'écriture du programme et le résultat des actions. C'est ce formalisme que l'on appelle un langage.

⇒ **Langages de 1^{re} génération :**

Les instructions du programme sont composées de codes binaires directement compréhensibles par le MC68000. C'est ce que l'on appelle le *langage machine*. Il est propre au microprocesseur. C'est pour cela, entre autres, qu'un programme destiné à un IBM ne fonctionne pas sur un ATARI ST, et vice-versa, les processeurs différant.

Les codes binaires sont incompréhensibles et difficilement manipulables par le commun des mortels. C'est pour cela qu'on a créé d'autres langages plus pratiques.

☞ **Remarque :**

Les tout premiers programmes sur les tout premiers ordinateurs furent écrits en langage machine.

⇒ **Langages de 2^e génération :**

Le **LANGAGE D'ASSEMBLAGE**, ou langage de bas niveau, est très proche du langage machine mais les codes binaires y sont remplacés par des symboles (mots) aisément compréhensibles et faciles à retenir. L'addition du contenu de deux registres, par exemple, est symbolisée par :

```
ADD r1,r2
```

On parle aussi de *langage orienté-machine*, car il dépend totalement des caractéristiques physiques du processeur: opérations de base, modes d'adressages, nombre de registres, etc. Le programmeur doit posséder une connaissance approfondie non seulement du processeur, mais aussi du système d'exploitation.

Les programmes écrits en assembleur ne sont pas du tout "portables", c'est-à-dire qu'il faut les réécrire entièrement si l'on veut les transporter sur une autre machine.

◊ **Remarque :**

Le tout premier assembleur (compilateur du langage d'assemblage) fut écrit en langage machine.

⇒ **Langages de 3^e génération**

Les langages évolués (BASIC, COBOL, LISP) apportent, par rapport à l'assembleur, un niveau d'abstraction à la programmation: il ne faut (presque) plus connaître la machine sur laquelle va tourner le programme que l'on écrit et l'on peut demander à l'ordinateur d'exécuter des tâches compliquées sans avoir à connaître les détails du comment de leur accomplissement. Imprimer un texte à l'écran, par exemple, se fera toujours en BASIC par une instruction du type :

PRINT texte

Ceci, quel que soit le matériel sur lequel on travaille. Cela augmente la portabilité des programmes.

On parle aussi de *langages de haut niveau* ou orientés-problèmes car ils sont destinés à résoudre une classe plus ou moins précise de problèmes:

mathématiques (FORTRAN), commerciaux (COBOL), symboliques (LISP). Certains se veulent plus généraux et permettent de traiter avec plus ou moins de bonheur presque tous les types de problèmes: BASIC, C, PASCAL...

⇒ Langages de 4^e génération

On regroupe sous ce vocable les générateurs de systèmes experts et les SGDB (Systèmes de Gestion de Base de Données). Ils prennent en charge une grande partie de la programmation, laissant à l'utilisateur l'analyse du problème.

Ils augmentent la productivité du programmeur. Etant prévus pour résoudre une classe générale de problèmes, ils ne sont pas toujours aussi performants (rapides) qu'un programme sur mesure, écrit dans un langage de génération inférieure.

☛ INTERPRETEUR / COMPILATEUR

Une fois le texte du programme composé dans un langage symbolique ou de haut niveau, quel qu'il soit, il faut le traduire en langage machine, le seul compréhensible par la machine. Il y a deux manières d'effectuer cette traduction : compiler ou interpréter.

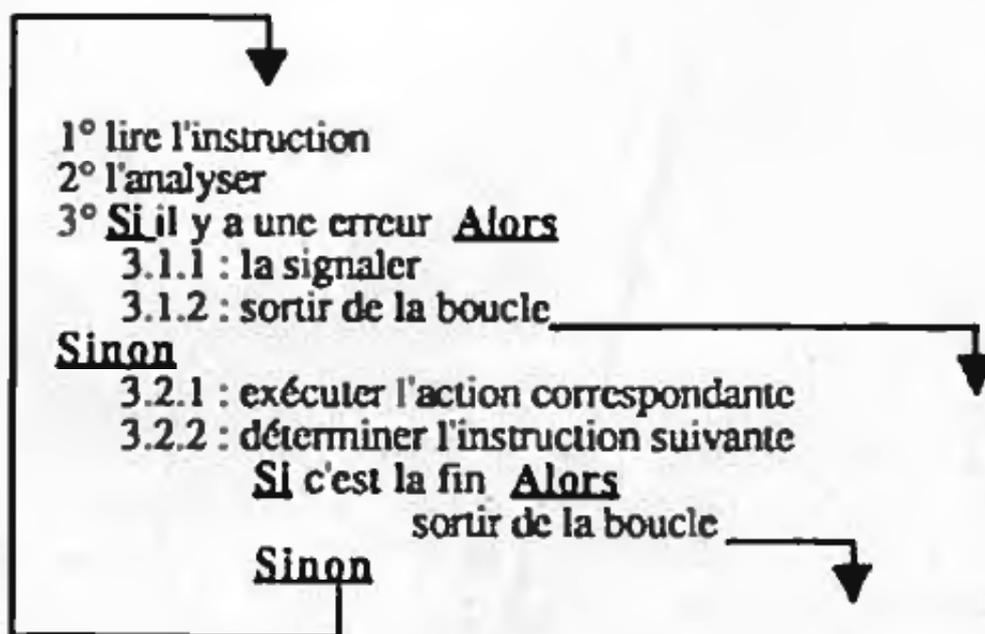
⇒ Un *compilateur* est un logiciel "*traducteur*". Il traduit le texte écrit dans un langage particulier (le *programme source*) en du code MC68000, directement exploitable par la machine (le *programme objet*). Cette opération - la compilation - doit être faite une seule fois.



Pour exécuter ce code, il suffira ensuite de le lancer comme tout programme usuel, sans plus avoir besoin du compilateur.

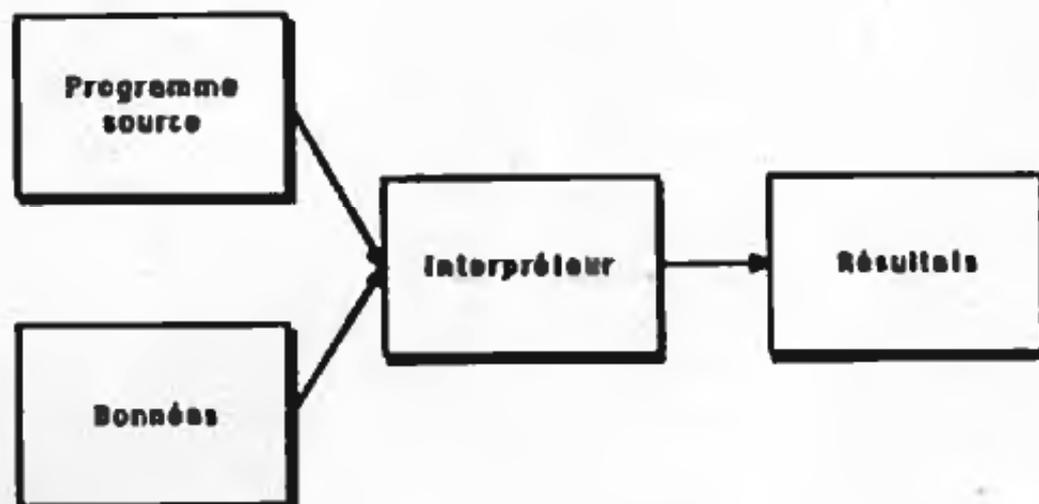


⇒ Un *interpréteur* ne crée pas de code objet pouvant tourner indépendamment. Après avoir été chargé en mémoire, le texte du programme source est parcouru et interprété dans l'ordre logique des instructions. On peut représenter l'interprétation par le schéma suivant:



Ce processus s'appelle une *boucle d'interprétation*.

Ne créant pas de programme objet capable de s'exécuter seul, l'interpréteur doit être présent lors de chaque exécution.



L'étape de traduction est localisée dans le 3.2.1. En réalité, il s'agit d'exécuter des routines pré-traduites de l'interpréteur correspondant aux différentes actions possibles. Elle est chaque fois précédée d'une analyse de l'instruction en langage de haut niveau. Cela signifie que si un morceau du programme doit être exécuté plusieurs fois, il sera à chaque passage analysé puis traduit de la même manière.

✎ **Remarques :**

✓ Certains interpréteurs BASIC sont disponibles avec un module "RUN-ONLY": une fois le programme exempt d'erreur, on lui associe le module, qui se charge de son exécution. On peut ainsi créer et diffuser

des applications prêtes à l'emploi (le module RUN-ONLY est une version courte de l'interpréteur, ne contenant que la partie s'occupant de l'exécution - RUN).

✓ Par rapport à l'exécution du code objet généré par la compilation d'un même programme source, l'interprétation est un processus beaucoup plus lent. Une fois le programme mis au point, on a intérêt, si un compilateur existe pour ce même langage, à le compiler.

✓ Un interpréteur s'avère cependant plus convivial lors de la mise au point de petits programmes, surtout pour les débutants. C'est cette raison qui a amené la création d'interpréteur C et PASCAL. Ces langages étaient, avant cela, réservés aux amateurs avertis.

✶ ÉDITEUR

Quel que soit le langage choisi, nous avons vu qu'un programme était avant tout un texte. Ce texte doit donc être écrit et sauvé dans un fichier qui sera ensuite interprété ou compilé. C'est le rôle de l'éditeur.

Deux cas peuvent se présenter :

◆ *L'éditeur est intégré au compilateur* (ou à l'interpréteur). Ces deux logiciels sont fusionnés en un seul et conçus pour travailler ensemble.

Avantages :

→ Disponibilité immédiate.

Une session de programmation est composée de nombreuses répétitions de la séquence "Edition du programme/Compilation". A chaque erreur détectée, il faut rééditer le texte, le corriger puis le recompiler. L'éditeur faisant corps avec le compilateur, il ne doit pas être rechargé en mémoire après chaque compilation : il est immédiatement disponible.

→ **Spécificité.**

Certains langages sortent tellement de l'ordinaire qu'un travail sérieux ne peut s'envisager qu'avec un éditeur spécialisé. LISP, par exemple, demande pour être lisible une mise en page précise, basée sur l'ouverture et la fermeture de parenthèses. Un éditeur en tenant compte s'avérera dans ce cas un atout important. APL emploie un jeu de caractères non standards et exige un éditeur spécialisé.

Désavantages :

→ **Perte de souplesse.**

Il est impossible d'utiliser votre éditeur favori, de choisir celui qui vous convient le mieux. De plus, si vous programmez dans plusieurs langages, il vous faudra apprendre les particularités et mode d'emploi de chaque éditeur intégré.

→ **Occupation de mémoire.**

Un ensemble intégré éditeur + compilateur est bien sûr plus volumineux qu'un compilateur seul. Cela peut poser problème à ceux qui possèdent une configuration mémoire limitée (520).

→ **Performances.**

Devant être compacts (voir ci-dessus), ils sont moins rapides et moins complets que les éditeurs individuels qui peuvent eux se permettre d'occuper plus de mémoire, devant travailler seuls.

♦ *L'éditeur n'est pas intégré.* Il faut alors disposer d'un éditeur individuel. Les avantages et désavantages correspondent respectivement aux désavantages et avantages des éditeurs intégrés.

Avantages :

→ Libre choix de l'éditeur.

→ Performances.

Le défilement de ligne (scrolling) du GEM n'est pas très rapide. Certains éditeurs (TEMPUS.PRG, par exemple) utilisent des routines personnelles pour l'accélérer.

Désavantages :

→ Chargement obligatoire avant toute utilisation.

→ Trop grande généralité.

o *Remarques :*

✓ Certains compilateurs (interpréteurs), lorsqu'ils détectent une erreur, lancent automatiquement l'exécution de l'éditeur - intégré ou non - et positionnent le curseur sur la ligne erronée. C'est ce genre de détails qui font toute la différence pour le programmeur. Le confort d'utilisation est irremplaçable lorsqu'il s'agit de faire un travail intensif et prolongé. Si un interpréteur ou compilateur ne se prévaut pas de cette caractéristique, c'est qu'il ne la possède sûrement pas.

✓ La plupart des traitements de textes peuvent servir d'éditeur. Il suffit de sauver le texte du programme sans aucun paramètre de mise en page ou de style, c'est-à-dire au format ASCII.

☛ **DEBUGGER**

Un debugger est un *utilitaire de programmation* que l'on emploie en association avec un compilateur. Il permet, une fois le programme au point, de surveiller ce qui se passe à l'intérieur de celui-ci lors de son exécution. On peut entre autres placer des points d'arrêts dans le programme. Arrivé au point d'arrêt, le programme s'arrête et donne le contrôle à l'utilisateur. On peut dès lors :

- ☛ examiner/modifier le contenu de la mémoire;
- ☛ examiner/modifier le contenu des registres;
- ☛ examiner/modifier le contenu des variables;
- ☛ être averti des changements de valeur;
- ☛ surveiller l'emploi de certaines routines;
- ☛ etc.

Le compilateur ne sait détecter que les fautes d'écriture par rapport aux règles syntaxiques du langage. Si une erreur de logique s'est glissée dans la conception du programme, il est parfois très difficile de la repérer. C'est pour cela que le débogueur est un outil irremplaçable dans les projets d'une certaine ampleur.

La quasi-totalité des debuggers ne sont compatibles qu'avec le compilateur de la même marque. On préférera parfois choisir un compilateur un peu moins performant qu'un autre mais accompagné d'un meilleur debugger.

☛ RESSOURCES

Nous avons vu que GEM offre au programmeur des facilités pour l'usage des menus, fenêtres et icônes. Il n'est cependant pas très facile de les utiliser et le débutant s'y cassera facilement les dents.

Les interpréteurs et compilateurs BASIC, pour la plupart, permettent heureusement d'aborder GEM d'une manière beaucoup plus aisée. L'utilisateur peut ainsi créer des menus ou manipuler des fenêtres sans devoir s'adresser directement au système.

☛ Remarque :

Il n'est pas obligatoire d'employer les ressources du GEM dans un programme. Il peut très bien être conçu simplement, sans fenêtre ni menus, comme sur les autres ordinateurs. Dans ce cas, on peut remplacer le suffixe .PRG par .TOS. L'utilisateur remarque ainsi directement qu'il s'agit d'un programme non-GEM.

☛ MÉLANGE DE LANGAGES

Le confort et la facilité apportés par les langages de haut niveau se fait parfois au prix d'une relative lenteur : le code généré par un compilateur est toujours moins rapide que celui écrit en assembleur par un bon programmeur. Cela peut parfois poser problème mais, heureusement, il existe un compromis: employer l'ASSEMBLEUR pour les parties du programme qui doivent être très rapides et écrire le reste

dans un langage de haut niveau. Ce mélange de langage peut se faire de deux manières distinctes :

○ Le compilateur *C Mégamax* possède un assembleur intégré. Il suffit d'insérer dans le texte même du programme source en C un morceau écrit en langage d'assemblage. Des repères indiquent au compilateur qu'arrivé à cet endroit il doit passer la main à l'Assembleur.

○ Les compilateurs *BCPL, C et Pascal de Metacomco* procèdent différemment. La partie de code à optimiser est développée séparément avec un Assembleur (de la même marque) qui produit un module séparé. Il sera ensuite relié (linked) au reste du programme.

L'Assembleur intégré est plus facile d'emploi, mais il est aussi moins performant qu'un Assembleur externe, pour les mêmes raisons que celles vues dans l'Editeur.

➔ POURQUOI DIFFÉRENTS LANGAGES ?

S'il existe une multitude de langages (± 400), c'est qu'il existe au moins une bonne raison. On peut en réalité en distinguer trois principales :

- ➔ spécificité du langage;
- ➔ récupération des programmes;
- ➔ l'habitude du programmeur.

◆ La spécificité du langage :

Selon le domaine auquel ils se rapportent - scientifique, gestion, intelligence artificielle - les problèmes à résoudre ont des besoins particuliers. Comme le langage universel n'existe pas, on emploie

dès lors celui qui répond le mieux aux besoins de l'application.

⇒ **La récupération des programmes :**

De nombreux problèmes possèdent déjà une solution. Adapter un programme existant qui répond à un problème similaire coûte moins cher qu'en réécrire un nouveau, même dans un langage beaucoup plus moderne. Ce désir, justifié, de conserver l'acquis explique que l'on vende encore actuellement des compilateurs FORTRAN ou PL/1, bien que ces langages soient tout à fait dépassés.

⇒ **L'imprégnation du programmeur :**

"The baby duck syndrom"

Comme le petit canard qui prend pour maman le premier objet qu'il voit en naissant (phénomène d'imprégnation), le programmeur s'attache sentimentalement aux premiers langages appris dans sa carrière : il leur pardonne bien des défauts (qui sont pour lui des "caractéristiques") et rechigne à les abandonner.

Après ces quelques remarques générales, examinons les différents langages disponibles sur ST.

2. LES DIFFÉRENTS LANGAGES :

Passons en revue les langages disponibles sur ATARI ST. Les moins connus du grand public (APL, OCCAM, LISP) seront d'abord présentés, avant d'examiner leur implémentation sur ATARI ST.

Dans notre description, nous suivrons l'ordre suivant:

- ⇒ L'ASSEMBLEUR : le bas-niveau.

- ⇒ C : } le moyen-niveau.
- ⇒ BCPL : }

- ⇒ FORTRAN : } les programmes "spaghetti".
- ⇒ BASIC : }

- ⇒ PASCAL : } la programmation structurée.
- ⇒ MODULA : }

- ⇒ LISP : l'Intelligence Artificielle.

- ⇒ APL : } le matheux.
- ⇒ FORTH : }

- ⇒ OCCAM : le petit nouveau.

☛ ASSEMBLEUR

☐ Abus de langage

L'ASSEMBLEUR n'est pas un langage, mais un compilateur du langage d'assemblage. Il transforme le code source - écrit en langage d'assemblage - en une séquence d'instructions en langage machine 68000.

☐ Sinécure

Non, n'y comptez pas trop. La programmation en ASSEMBLEUR n'est pas et ne sera jamais une sinécure. Elle vérifie toutes les lois de Murphy. Vous devez penser à tout, gérer tout, tout réinventer, ou presque. Un homme normal ne programme en ASSEMBLEUR que s'il y est vraiment forcé.

☐ Atouts

Darwin vous le dirait lui-même: si l'ASSEMBLEUR survit malgré son inconfort notoire c'est que, quelque part, il est le plus fort. Oui, c'est vrai, il faut au moins lui reconnaître cela. En compacité et en vitesse, il est imbattable. Lorsqu'on dispose d'un 520 ou d'un 1040, la compacité perd un peu de son intérêt. On l'utilisera donc surtout pour les tâches critiques qui ne souffrent aucune lenteurs.

☐ Macro-Assembleur

Tous les produits vendus sur le marché sont en réalité des macro-assembleurs. Au langage d'assemblage standard du MC68000, ils ajoutent un ensemble de facilités: labels, pseudo-instructions, assemblage conditionnel, macro-instructions.

Une *pseudo-instruction* est une instruction destinée à l'ASSEMBLEUR. Une macro-instruction est une no-

tation raccourcie d'un bloc d'instructions en langage d'assemblage.

□ Debugger

Ce qui s'écrit en 10 lignes dans un langage de haut niveau peut facilement en demander des centaines en ASSEMBLEUR. C'est surtout vrai si vous faites appel aux routines du GEM et du TOS. Dans de telles conditions, un debugger est une nécessité. Il doit être puissant et confortable et, plus que tout autre caractéristique, c'est lui qui doit orienter le choix d'un ASSEMBLEUR.

□ Assembleur croisé

On en distingue deux types.

Le premier type permet de développer sur votre ATARI ST un programme en langage d'assemblage qui tournera sur une autre machine à base de 68000, comme l'Amiga ou le Macintosh.

Le second permet de développer, toujours sur votre ATARI ST, un programme dans un autre langage d'assemblage que le 68000, c'est-à-dire destiné à tourner sur un autre micro-processeur. C'est le cas du *Memocom Cross-16 Meta Assembler*.

C

□ Histoire

C fut créé et développé à la fin des années 70 par Dennie Ritchie à partir du langage B. Il sert à écrire le système d'exploitation *Unix* pour le mini-ordinateur DEC-PDP11.

□ C comme compact

Le langage C est "pauvre". Cela facilite à la fois son apprentissage et la construction de compilateurs. Il utilise les mêmes types d'objet que le processeur : adresse, caractère, nombre. Il ne dispose par contre d'aucune opération prédéfinie pour manipuler les objets composés - chaînes de caractères, vecteurs - comme c'est le cas dans la plupart des autres langages.

□ Son entourage

Simplicité ne veut cependant pas dire limite. Il compense le caractère minimal de son noyau de base en s'entourant de nombreuses bibliothèques de routines. On peut dire que c'est un langage de moyen niveau. Entre l'ASSEMBLEUR et les langages évolués. Il garde la souplesse de l'un en profitant des avantages des autres. Comme preuve de sa puissance remarquons que sur les 13000 lignes du code-système d'Unix seules 800 sont écrites en ASSEMBLEUR.

□ Le standard

La référence est le livre "*Le langage C*" de *W. Kernighan et D. Ritchie*. La norme ANSI, apparue récemment, précise sa formalisation et l'étend.

□ C et ST

Si la programmation de l'ATARI ST est tellement liée au langage C, c'est principalement parce que, d'une part, le GEM a été écrit presque totalement en C et, d'autre part, parce qu'il fut le premier langage de haut niveau disponible pour les développeurs. L'accès aux routines du GEM (VDI + AES) est fourni très simplement par l'adjonction de bibliothèques *ad hoc*.

□ Compilateur C

Ils sont légions. Tous - ou presque - suivent totalement le "*Kernighan Richie*". Certains (*Lattice C* par exemple) offrent les extensions nécessaires à la norme ANSI. Ils se distinguent principalement par les points suivants :

- ✓ 1° les bibliothèques de fonctions accompagnatrices;
- ✓ 2° la taille et la vitesse d'exécution du code compilé;
- ✓ 3° la vitesse de compilation;
- ✓ 4° l'existence de debugger symbolique.

□ Interpréteur C

Il existe mais n'en attendez pas des merveilles. Comme tout langage interprété, il est plus lent que la version compilée et exige toujours la présence de l'interpréteur pour exécuter un programme. Son interactivité le destine surtout à l'apprentissage du langage.

BCPL

Au début était le CPL : *Combined Programming Language*. Développé à l'université de Cambridge, il y fut condensé en BCPL - *Basic CPL* - qui fut condensé à son tour en B. Celui-ci fut développé et devint le C. Mais revenons-en au BCPL.

Si le C vous paraît trop restrictif, et l'ASSEMBLEUR pas assez structuré, c'est peut-être le langage que vous attendiez. A mi-chemin entre les deux, il combine la souplesse et la puissance du premier (on peut tout faire avec la machine) à la structuration et aux facilités de haut niveau du second. En ce qui concerne le "typage" des variables, il est encore moins restrictif que le C. Il les connaît sous forme de doubles-mots (32 bits) où l'on peut indistinctement tout placer et tout manipuler: pointeur, entier, caractères, booléens, etc.

La version vendue par Metacomco apporte quelques extensions au langage originel: opérations sur les réels (floating point) et conversion entier <-> réel. On peut aussi l'interfacer à des morceaux de codes écrits en ASSEMBLEUR ou en C (de la même marque).

☛ FORTRAN

FORmula TRANslation.

C'est le premier, le tout premier langage de haut niveau de l'histoire de l'informatique. Plutôt destiné aux gros systèmes, sa présence sur ATARI ST est plutôt anecdotique. Disons simplement qu'il existe un compilateur "FORTRAN 77" pour ST (*Prospero Fortran*) et qu'il permet d'accéder au GEM. Le seul intérêt de ce langage est de pouvoir récupérer d'anciens programmes. Comme son nom l'indique, c'est un matheux. Cela ne signifie pas ici qu'il est particulièrement doué pour cela, mais plutôt que c'est la seule chose qu'il sait faire correctement.

☛ BASIC

□ Le début

Beginners All-purpose Symbolic Instruction Code. Il a été créé en 1964 par John Kemeny et Thomas Kurtz spécialement pour les non-spécialistes. Au début de la micro-informatique, sa simplicité en faisait le langage idéal car il se contentait des mémoires exigües d'alors.

□ Les nouveaux Basic

GFA Basic, Fast Basic, Omikron Basic, Real Basic, etc. Ils ont tous adoptés la programmation structurée et le look de PASCAL : fonctions, procédures, passages de paramètres, modularité des programmes, disparition de numéros de lignes et des GOTO (finis les spaghettis). Il est enfin possible de faire des programmes "sérieux" et performants en BASIC. Ils tournent vite et bien. Si cela ne vous suffit pas, compilez-les. C'est rapide et pas cher.

□ Basic ANSI

Il faut signaler l'existence d'une norme ANSI en BASIC: le TRUE BASIC. Kemeny et Kurtz ont amélioré leur vieux standard et l'ont installé sur une large gamme de micros dont Amiga, Macintosh, AppleII et IBM PC. S'il n'atteint pas, loin s'en faut, les performances des BASIC ST cités plus haut, il présente néanmoins l'intérêt de sa norme : un programme en TRUE BASIC développé sur ST tournera aussi bien sur les autres machines. Il offre à cet effet des fonctions graphiques indépendantes du type et de la résolution du moniteur employé. La manipulation des menus, fenêtres, souris, dialogues et alertes est aussi grande-

ment facilitée par un jeu de fonctions spécialisées, non standard mais très simples d'usage.

□ **Les spécialistes**

A côté des dialectes à usage général existent deux versions développées dans un but précis.

"*Stos, the Game Creator*". Il offre 320 commandes destinées à simplifier la création de jeux d'actions rapides, où se mêlent sprites et musique.

"*BasALG*" : c'est un produit français distribué par Pressimage. Lui, ce serait plutôt les maths : fonctions trigonométriques, hyperboliques, intégrales, dérivées, manipulation de matrices, calcul en multi-précision, etc.

☛ PASCAL

Développé en 1970 par Niklaus Wirth, professeur à l'ETH de Zurich, le langage PASCAL tire son nom du philosophe et mathématicien français Blaise Pascal qui inventa la première machine à calculer.

Développé initialement pour l'enseignement de la programmation structurée, il a vu sa popularité s'accroître rapidement dans bien d'autres milieux.

De son origine pédagogique, PASCAL tire ses points forts - il encourage l'écriture de programmes clairs, structurés, faciles à lire et à maintenir - et ses faiblesses - il est rigide et ne manipule que des structures simples. Les fichiers, par exemple, ne peuvent être que séquentiels.

□ ISO

Pour rendre leur produits plus attractifs, les fabricants de compilateurs ont tous introduit leurs propres extensions au langage, créant une multitude de dialectes strictement incompatibles entre-eux. C'est pour assurer la portabilité des programmes entre différents compilateurs et créer un langage commun que la norme ISO Standard Pascal fut introduite en 1982.

□ Extensions standards

Malheureusement, la norme ISO n'autorise pas l'emploi des caractéristiques propres à l'ordinateur. Il est pourtant parfois souhaitable de pouvoir y accéder. Les compilateurs que l'on trouve sur ATARI ST apportent tous leur lot d'extensions "standards".

□ Exemples

Le *MCC Pascal* (de Metacomco) et le *Prospero Pascal* sont tous deux "validés ISO". Cela signifie que tout programme qui respecte strictement la norme ISO est assuré d'une compilation complète et correcte. Ils permettent de plus d'accéder aux routines du GEM (VDI + AES). Un bonus du MCC est la compatibilité avec les autres produits Metacomco : on peut, à partir d'un programme PASCAL, appeler des sous-routines écrites en C ou en ASSEMBLEUR. Le Prospero possède un éditeur intégré. Tous deux fonctionnent en deux modes : ISO strict, et ISO + extensions. Le premier mode permet de vérifier que vos programmes suivent la norme ISO.

□ Interpréteur

Signalons pour terminer l'existence d'un interpréteur Pascal pour ST. Son intérêt est surtout pédagogique.

MODULA-2

□ Histoire

Ce langage est le fruit de l'évolution de ses ancêtres directs : PASCAL et MODULA. Ils possèdent un air de famille: le même père (N. Wirth) et le respect de la programmation structurée.

Examinons-les succinctement.

PASCAL a été créé pour répondre à des besoins généraux de programmation. Il ne se voulait pas spécifique à un type de tâches particulier.

MODULA est un *langage expérimental*. Il n'a jamais dépassé le cadre du laboratoire. Il fut implanté pour la première fois en 1975.

Faut-il vraiment vous dire que MODULA-2 est le successeur de MODULA ? Non ? Alors ne le disons pas. Contentons-nous de préciser que sa première implémentation date de 1979 et qu'il possède toutes les caractéristiques de PASCAL avec en plus :

✓ *Une syntaxe plus homogène :*

En particulier, toutes les instructions qui commencent par un mot clé se terminent par un mot clé.

Exemple :

```
WHILE .. DO .. END
IF .. THEN .. ELSE ...END
```

Cette homogénéité facilite l'apprentissage et rend les programmes sources plus lisibles.

✓ *La modularité :*

Ce principe s'inspire du *langage Mesa*. Un programme peut être divisé en plusieurs modules, chacun étant compilé séparément. Cela permet de récupérer le travail fait antérieurement (on ne doit pas réinventer la roue à chaque fois) et d'accélérer les sessions de travail.

✓ *Un module peut de plus être divisé en deux parties: définition et implémentation.*

La partie définition spécifie tout ce que le module fait et ce que l'on connaît de lui. Elle suffit à l'emploi du module. Il ne faut pas en savoir plus. La partie implémentation est le corps du module, qui exécute vraiment tout ce qui est promis dans la définition.

✓ *Les facilités de bas niveau :*

Les langages de haut niveau sont caractérisés par l'abstraction qu'ils introduisent dans la programmation. Cela permet d'avoir une vue conceptuelle du problème, sans se préoccuper des caractéristiques propres à la machine et de dépister plus facilement certaines erreurs. Si l'on manipule des nombres entiers, par exemple, il ne faut pas savoir comment l'ordinateur les représente. A ces facilités correspondent des règles strictes à respecter : on ne peut placer un réel dans une variable destinée aux nombres entiers. Il existe deux cas où ces règles deviennent de véritables entraves et limitent les possibilités de programmation :

1. S'il faut exploiter des données générées par un autre ordinateur, employant une autre représentation des données.

2. Si l'on veut accéder aux caractéristiques internes du système.

Grâce aux facilités de bas niveau, le programmeur peut résoudre ces problèmes sans être obligé de recourir au langage machine, ce qui est toujours assez lourd.

✓ *Le type procédure :*

Une fonction ou une procédure peut être assignée à une variable.

✓ *Les processus :*

Ils donnent accès à la multi-programmation.

LISP

A constater le "boom" que connaît l'IA (Intelligence Artificielle), on pourrait croire que le LISP est un langage assez récent. Et bien non, il n'en est rien. Le LISP (LISt PRocessing) est même, juste après FORTRAN, le plus vieux langage de haut niveau qui soit. Il fut développé au MIT (Massachusetts Institute of Technology) par John McCarthy à la fin des années 50, époque où les tubes à vides cédaient la place aux premiers transistors.

▷ Principes

Les bases de LISP sont très simples, presque rudimentaires. LISP est interactif : l'interpréteur évalue directement ce qu'on lui soumet. La seule structure manipulée est la paire d'objets. En chaînant des paires entre elles, on obtient une liste, d'où le nom du langage. A partir de quelques fonctions très simples de manipulation de ces listes, on peut étendre le langage à volonté. C'est cette souplesse d'adaptation et la puissance qui en résulte qui ont fait de LISP "le" langage de l'IA. S'il est plus difficile à mettre en oeuvre que PROLOG, l'autre langage de l'IA, il est cependant plus souple et plus général.

▷ Domaine d'application

Son champ d'action couvre bon nombre d'applications difficilement traitables par les langages procéduraux (C, BASIC, PASCAL.) classiques : systèmes experts, robotique, compréhension du langage naturel, calcul symbolique.

La compréhension du langage naturel permet de créer des interfaces hommes-machines

autorisant l'utilisateur à communiquer avec l'ordinateur dans sa langue maternelle.

Le calcul symbolique est opposé au calcul numérique. L'expression " $x^2 + 2xy + y^2$ ", en calcul numérique sera évaluée directement et fournira un résultat numérique si les variables x et y possèdent une valeur, ou une erreur dans le cas contraire.

Considérée symboliquement, cette même expression est vue comme une somme de produits et de puissances de variables. Un programme LISP pourra très facilement la factoriser ou la dériver.

▷ LISP et ST

Le langage LISP, sur ATARI ST, est bien représenté. A côté des interpréteurs appartenant au domaine public, dont le XLISP de D. Betz, existent plusieurs produits commerciaux de très bonne qualité. Le *Cambridge LISP de Metacomco*, par exemple. Il appartient à la famille des "*Standard Lisp*", avec quelques extensions. Il possède plus de 400 fonctions "standard", sans parler de celles qui donnent accès au GEM. Il supporte les entiers, réels double-précision, rationnels (forme fractionnelle de stockage des nombre; exemple : 1/3), fonctions trigonométriques, fichiers, etc. Il comprend en outre un compilateur intégré. La mise au point d'une fonction se fait en mode interprété. Une fois correcte, elle est compilée afin d'accélérer ses usages futurs.

Il est difficile d'expliquer le charme de la programmation LISP à ceux qui ne l'ont jamais pratiquée ou en sont resté aux premiers balbutiements. Comme APL, FORTH et PROLOG, il s'agit d'un langage hors du commun, qui peut exiger une autre forme de pensée, ne le rendant pas accessible à tous. Comme eux, il en-

voûte ou il dégoûte, mais jamais il ne laisse indifférent. Son apprentissage n'est pas aisé, mais il est rentable. Une fois maîtrisé, il vous permettra de créer très facilement des programmes efficaces qui seraient difficilement réalisables dans un autre langage.

☛ APL

A observer la diversité des tâches soumises aux ordinateurs actuels, on en arrive parfois à oublier que leur fonction première était uniquement de faire des calculs ("*computer*" = calculateur). C'est dans ce contexte, à la fin des années soixante, que Kenneth Iverson, un mathématicien d'Harvard, mit au point un système de notation symbolique des algorithmes mathématiques. Il le décrit dans un livre - *A Programming Language* - qui sert encore de référence. Le premier interpréteur APL vit le jour en 1965 dans les laboratoires d'IBM.

La puissance et la concision de ce langage n'ont d'égal que son érotisme. La première rencontre avec un programme APL est toujours déconcertante, car il ne se raccroche à rien de connu.

A mi-chemin entre un extrait du *Grand Albert* et un ouvrage de mathématiques appliquées, APL est composé d'une cinquantaine de fonctions. Chacune d'elles est représentée par un symbole unique: lettre grecque (rho, iota), graphique (cercle, rond, triangle, flèche), opérateur classique (+, -, *, /, =, <, ...) ou non (virgule), etc. Ces symboles peuvent s'employer de deux manières différentes, monadique ou diadique, suivant qu'on leur fournit un ou deux arguments. La fonction "iota" par exemple avec un argument "n" génère la suite de nombre de 1 à n. Avec deux arguments, elle cherche dans la liste de droite le nombre situé à gauche et en renvoie la position.

Les fonctions APL concentrent donc en un seul caractère ce qui s'écrirait en des dizaines voire des centaines de lignes dans tout autre langage: division de matrices, tri, recherche d'élément dans une inversion/transposition de matrice, conversion d'un nombre d'une base en une autre, tirage au hasard de n nombres parmi une suite donnée (comme distribuer des cartes), concaténation de deux structures, création d'une matrice de dimension quelconque, etc..

Beaucoup de ces fonctions sont "orthogonales": elles peuvent s'appliquer indifféremment à des éléments d'un type quelconque :

Exemple : $A \leftarrow B + C$

La variable A reçoit le résultat de l'addition de B et C qui peuvent être des nombres, des caractères, mais aussi des vecteurs ou des matrices à plusieurs dimensions. On ne sera pas étonné d'apprendre que le *crible d'Erathostène* (programme produisant les nombres premiers entre 2 et N) peut s'écrire, en APL, en une seule ligne.

APL est un langage interprété. Il peut s'utiliser de deux manières distinctes.

En *mode direct*, il travaille comme une super-calculatrice, affichant directement le résultat des calculs soumis.

En *mode définition*, un éditeur intégré permet de mettre au point et de stocker les fonctions souvent employées ou de créer des programmes complexes.

L'affichage et l'accès direct à tous les symboles spécifiques à l'APL exigent des terminaux dédiés. L'écran monochrome de l'ATARI ST est idéal pour cette tâche et, en ce qui concerne le clavier, l'APL 68000 de

Spencer Organization est fourni avec un jeu d'autocollants à coller sur le devant des touches. Il permet aussi l'accès à toutes les fonctions du GEM (fenêtres, menus, graphiques) et la manipulation aisée des fichiers.

☛ FORTH

Conçu à l'origine pour piloter un télescope, ce langage s'est étendu par la suite à bien d'autres domaines, orientés principalement vers les mathématiques.

☞ Principe

L'essence de la programmation FORTH est de diviser tout problème en petits morceaux. FORTH étant un environnement interactif (interpréteur), ils sont faciles à créer et à tester. Chaque nouveau morceau développé constitue un "*mot*" qui devient partie intégrante du dictionnaire FORTH, au même titre que les autres. On pourra l'employer en mode immédiat - au clavier - ou dans un programme quelconque, simplement par l'appel de son nom.

Les mots de base du dictionnaire comprennent de nombreuses structures de boucle (DO/LOOP, BEGIN/AGAIN, BEGIN/UNTIL, BEGIN/WHILE/REPEAT) et de test (IF/THEN/ELSE, CASE).

Ajoutons que FORTH pratique l'arithmétique en notation polonaise inversée (NPI) comme le font les calculatrices Hewlett-Packard. La NPI, aussi appelée *notation postfixée*, consiste à placer dans une expression arithmétique l'opérateur après (post) les opérandes.

$(1 + 2) * (3 - 4)$ devient $1 2 + 3 4 - *$.

☞ FORTH et ST

A côté des nombreux environnements FORTH disponibles dans le Domaine Public existent des produits commerciaux plus riches, GEM-FORTH et FORTH/MT, par exemple.

Ils permettent de développer des programmes exécutables séparément et d'intégrer du code Assembleur. Ils apportent l'accès à toutes les fonctions du GEM et de l'OS, ainsi que la manipulation des nombres réels sur 32 bits, les fonctions trigonométriques, logiques et de manipulation de pile.

Ils se vantent même de capacités multi-tâches. Ne nous méprenons pas : il n'est pas possible de lancer en parallèle plusieurs programmes FORTH. On peut seulement, dans un même programme, développer des morceaux de code qui tourneront en pseudo-parallélisme. Un gestionnaire de tâches (*tasks scheduler*) intégré au programme principal permet à chaque morceau - tâche - de s'exécuter pendant un court instant après quoi un autre reçoit le tour.

Destiné surtout à la programmation scientifique et mathématique, FORTH offre à la fois les facilités des langages de haut niveau (dictionnaire incrémental) et la rapidité alliée à la souplesse de l'ASSEMBLEUR. Sa philosophie particulière (dictionnaire, NPI, stack) l'éloigne des langages structurés habituels et son apprentissage peut demander, de ce fait, un effort plus important.

☛ OCCAM

L'ATARI ST dispose d'un seul micro-processeur : le MC68000. Il exécute séquentiellement les instructions en langage machine de nos programmes (une par une).

□ Ordinateur parallèle

Nous verrons dans le chapitre consacré aux *PÉRIPHÉRIQUES* que le ST peut servir d'hôte à l'ABAQ. En quelques mots, l'ABAQ est un ordinateur qui est construit autour de un ou plusieurs micro-processeurs centraux: les TRANSPUTER, d'Inmos. On appelle cela un ordinateur parallèle. Quel avantage y a-t-il à disposer de plusieurs micro-processeurs au lieu d'un seul ? Prenons un exemple qui en montre les avantages.

Soit le programme BASIC suivant :

```
FOR i 1 TO 10
  A(i) = B(i) + C(i)      (1)
NEXT
```

La ligne (1) sera exécutée 10 fois de suite, pour i variant successivement 1, 2, ... 10. On peut remarquer que chacune de ces dix exécutions n'est aucunement influencée par les neuf autres. Elles sont totalement indépendante. On pourrait les exécuter dans le désordre que le résultat n'en serait pas changé. Si l'ordre n'a pas d'importance, on peut aussi les exécuter en même temps, sur 10 processeurs différents. Le résultat sera le même, mais cela ira (presque) dix fois plus vite.

Un autre exemple typique est:

$$A = (B * C) + (D * E)$$

Les calculs entre parenthèses peuvent être exécutés dans n'importe quel ordre.

Le parallélisme est donc une bonne chose. Mais comment le programmer ?

□ Un langage parallèle

Il est possible de programmer l'ABAQ en C, FORTRAN, BCPL, LISP, PASCAL. Ces compilateurs doivent alors détecter eux-même les opérations parallélisables (exemple: $(1 + 2) * (3 + 4)$) pour en accélérer l'exécution. Cette tâche est très difficile, et les programmes objets générés sont rarement optimaux. De plus, étant par essence séquentiels, ces langages de haut niveau ne permettent pas au programmeur de gérer explicitement le parallélisme. Il fallait donc un nouveau langage.

□ OCCAM

C'est dans ce but que OCCAM a été développé. Ce nom bizarre n'est pas un acronyme, mais celui de William of Occam, philosophe du XIV^e siècle. Les informaticiens aiment faire référence au passé (Pascal, Ada).

Un programme OCCAM est composé de blocs d'instructions devant être exécutées séquentiellement. Chaque bloc - ou processus - peut contenir un ou plusieurs autres bloc.

Le parallélisme : les blocs d'un même niveau peuvent évoluer en parallèle. Idéalement, chacun sera exécuté sur un processeur différent.

La communication : des processus évoluant parallèlement peuvent échanger de l'information à travers des canaux de communication.

Pour écrire un caractère à l'écran, un processus doit l'envoyer via le canal "screen" de la manière suivante :

screen ! car

Pour lire un caractère au clavier, il faut le lire sur le canal "keyboard":

keyboard ? car

La communication entre processus se fait de la même manière.

⇒ Un exemple

Soit un programme où trois processus évoluent en parallèle.

Sa structure de base sera :

PAR

SEQ

instructions du processus 1

SEQ

instructions du processus 2

SEQ

instructions du processus 3

□ K-OCCAM

Les ABAQ sont rares et chers. Il est cependant possible, grâce à K-OCCAM, de créer des programmes en OCCAM et de les faire tourner sur un ATARI ST standard, muni de son seul MC68000. Bien sûr, il seront plus lents que sur des *Transputer*, mais vous pourrez ainsi satisfaire votre curiosité ou préparer votre logithèque pour le jour où...

K-OCCAM est constitué de trois parties : un éditeur, un compilateur, un interpréteur.

L'*éditeur* vous permet de composer le code source en Occam.

Le *compilateur* le traduit en un langage d'assemblage intermédiaire - le "*Portakit*" - qui ressemble assez au langage machine du Transputer d'Inmos.

L'*interpréteur*, enfin, lit le code intermédiaire, émule les instructions en *Portakit* en fonction du processeur hôte (MC68000) et en répercute les résultats. Le parallélisme présent dans un programme OCCAM est simulé par l'interpréteur en faisant du "time-slicing": chaque processus peut disposer du processeur pendant une "tranche" de temps, après quoi un autre reçoit le tour, et ainsi de suite jusqu'à la fin du programme. Cette émulation est transparente à l'utilisateur qui a vraiment l'impression de disposer de plusieurs processeurs, si ce n'est que cela tourne beaucoup plus lentement.

Remarques :

✓ Il est impossible d'accéder au GEM et à ses facilités à partir de K-OCCAM. La gestion de fichiers est assez minimale.

✓ Les programmes écrits avec K-OCCAM respectant la norme OCCAM, il suffit d'un compilateur OCCAM pour les faire tourner sur *Transputer*.

□ Conclusions

K-OCCAM ne peut pas servir à créer des programmes professionnels; ce n'est qu'un outil didactique. Il permet d'aborder, à peu de frais, le monde de la programmation parallèle. Ce n'est déjà pas si mal.

LE GRAPHISME





Le graphisme est un des domaines de prédilection de l'ATARI ST. Avant de parler des logiciels qui permettent de "faire" du graphisme, examinons l'aspect matériel que cela comporte.

1. LE MATERIEL

☛ L'équipement

L'équipement de base de la gamme ST permet de raccorder tous les modèles (du 520 ST au méga ST) à un moniteur soit couleur soit monochrome. Il suffit de raccorder l'écran de votre choix pour que le système reconnaisse automatiquement son type. Pas de carte vidéo à rajouter. Les modèles 520 ST sont de

plus équipés d'un modulateur vidéo grâce auquel on peut les raccorder directement à une télévision.

☛ Les couleurs

Le générateur vidéo est capable de produire 512 couleurs différentes, du noir au blanc, formées à partir des trois fondamentales: rouge, vert et bleu. On ne peut cependant en afficher au maximum que 16 à la fois. Certains logiciels permettent cependant de dépasser cette limite et offrent simultanément 512 couleurs distinctes à l'écran, ou même plus (voir plus loin la partie "*logiciels*"). Lorsqu'on appelle l'accessoire panneau de contrôle, on y trouve trois curseurs permettant de composer des couleurs en dosant les quantités de rouge, vert et bleu qu'elles contiendront. Ce dosage comporte huit graduations, allant de 0 - absence de couleur - à 7 (couleur pleine). N'importe laquelle des 512 couleurs disponibles peut être représentée de la sorte par un nombre de trois chiffres octaux. Le rouge pur est 700, le vert pur 070, le bleu pur 007, le blanc 777 et le noir 000. Un chiffre octal (de 0 à 7) se code avec trois bits. $512 \text{ couleurs} = 8 \times 8 \times 8$. Donc, $3 \times 3 \times 3 = 27$ bits sont nécessaires pour désigner une couleur avec précision.

L'indispensable pour Atari

Matériel et logiciels

Atari est un micro-ordinateur très largement répandu en France dont il existe plusieurs modèles qui se différencient uniquement par la taille de la mémoire centrale.

Le micro-ordinateur Atari est, à l'heure actuelle, celui qui présente le meilleur rapport qualité/prix. La chaîne de micro-édition construite autour d'un Atari est moitié moins chère qu'une chaîne équivalente configurée autour d'une autre marque.

En utilisant certains programmes, on peut rendre cet ordinateur IBM compatible et aussi Macintosh compatible.

Ce livre, véritable petite bible de l'Atari tant du point de vue du matériel que des logiciels, s'adresse bien sûr à tous ceux qui possèdent cet ordinateur, mais aussi à ceux, professionnels ou non, qui s'interrogent sur toutes ses possibilités.

L'auteur:

A. Ravet est licencié en informatique et possède depuis plusieurs années un Atari qu'il a soumis à tous les tests ludiques et scientifiques.

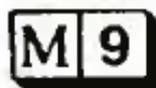
40 3861 8



9 782501 010948



marabout
SERVICE



☛ Les trois résolutions

La partie du TOS qui gère tout ce qui touche au graphisme sur l'ATARI-ST est capable de travailler dans trois résolutions différentes: la haute, la moyenne et la basse.

⇒ La haute résolution (HRG) :

Dans ce mode, seules deux couleurs sont disponibles: le noir et le blanc. L'une et l'autre peuvent indifféremment servir de couleur de fond ou de trait. La résolution est de 400 lignes de 640 points. En affichage de texte, cela permet d'avoir 25 lignes de 80 caractères, chaque caractère étant formé dans une matrice de 16 x 8 points. Cette grande finesse de détails, ainsi que la stabilité parfaite de l'image, obtenue par un rafraîchissement (redessin) de l'écran 70 fois par seconde assurent un confort de vision maximal.

⇒ La moyenne/basse résolution (MRG/BRG):

Elle offre 4/16 couleurs différentes à choisir parmi 512 et une résolution de 200 lignes x 640/320 colonnes. Le texte s'affiche en 25 lignes de 80/40 caractères, construits dans une matrice de 8 x 8 points. La basse résolution est surtout réservée au jeux ou au graphisme pur.

☛ Remarques :

- ✓ La haute résolution exige un moniteur monochrome. Il n'est pas

possible d'afficher une image noir et blanc sur un écran couleur.

✓ Il existe des programmes qui permettent, dans une certaine mesure, de faire tourner sur un écran couleur un programme ne travaillant qu'en haute résolution, ou un programme couleur sur un écran monochrome.

✓ Le fait de changer de type de moniteur en cours de session de travail déclenche un *reset* automatique du système. Cela préserve le moniteur couleur de la destruction s'il venait à recevoir le 70 Hz du mode monochrome.

☛ L'écran graphique

La plupart des ordinateurs gèrent séparément le graphisme. Ils stockent le texte dans une partie de la mémoire et c'est le circuit vidéo qui construit son image à partir d'une police de caractères unique. Le graphisme, quant à lui, est géré séparément dans une autre partie de la mémoire. L'ATARI-ST, au contraire, traite les deux cas de la même manière.

➤ *Avantage*: simplicité et souplesse. Il est très facile de mélanger texte et graphisme, comme d'afficher simultanément plusieurs polices de caractères de style et/ou de taille différents.

➤ *Inconvénient*: une certaine lenteur dans le défilement de texte (scrolling). Elle est due à la conception non optimale des routines du système qui gèrent cette tâche bien précise. Ce défaut peut être corrigé par

l'adjonction d'un coprocesseur graphique (*blitter*) ou par une réécriture de la routine *ad hoc*. C'est cette seconde solution qui a été choisie par les concepteurs de certains traitements de texte (*Le Rédacteur*, *Tempus*) réputés pour leur scrolling instantané.

L'image visible par l'utilisateur est une mosaïque de points colorés. Cette analogie est valable, dans une certaine mesure, pour expliquer la manière dont est géré l'écran graphique. Il occupe dans l'ordinateur une zone contiguë de 32 Ko, appelée **mémoire graphique**, où est stockée l'image qui apparaît sur le moniteur. Sa taille est la même pour les trois résolutions, mais elle est utilisée différemment selon le cas. Au démarrage du système (*boot*), la zone mémoire réservée à l'écran se trouve tout au bout de la RAM. L'adresse de la base de l'écran varie donc suivant la taille de cette RAM.

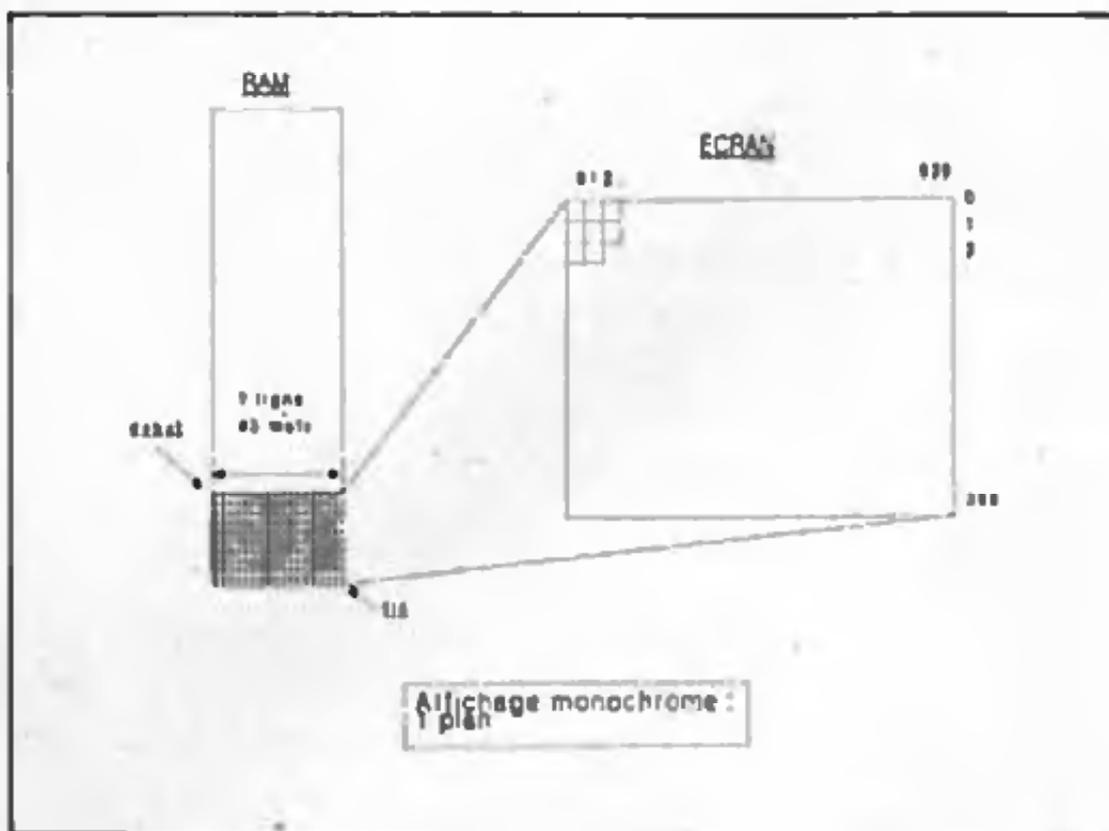
☛ Le principe du bitmap et de la palette

➔ Le principe bitmap

⇒ *Haute résolution :*

Il y a sur l'écran $640 \times 400 = 256000$ pixels (points) soit noir soit blanc. Chacun est représenté en mémoire par un bit qui représente sa couleur. Comme il n'y a que deux choix possibles, un seul bit suffit. Les pixels sont codés en mémoire ligne par ligne, le 1^{er} point en haut à gauche de l'écran étant situé au début de la mémoire graphique, et le dernier en bas à droite étant

à la fin. 640 lignes de 400 pixels points = 256.000 bits = 32.000 octets. Ce principe de mosaïque où tout point de l'écran est associé à un bit de la mémoire graphique est appelé *bitmap*. Le générateur graphique envoie au moniteur l'image, ligne par ligne, du contenu du bitmap.



⇒ *Moyenne résolution :*

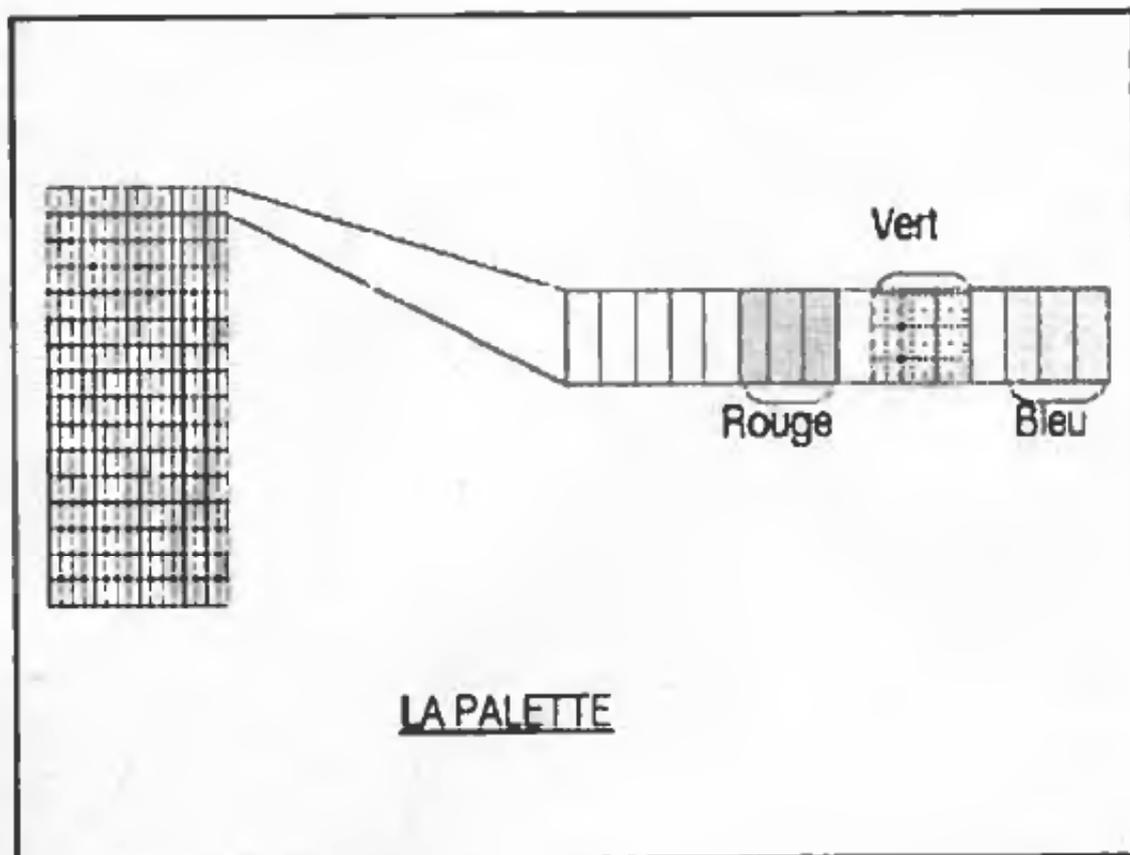
640 x 200 pixels. Quatre couleurs à choisir parmi 512. Nous avons vu qu'il fallait neuf bits pour identifier univoquement une couleur. Or on sait d'avance que le choix devra se faire parmi quatre couleurs préchoisies. On emploiera pour cela une palette de couleurs et un bitmap en deux plans :

→ La palette

Tout comme la palette du peintre où il prépare les quelques couleurs dont il va se servir, celle dont il s'agit ici contient les 4 ou 16 couleurs qui sont réellement utilisées (qui apparaissent à l'écran).

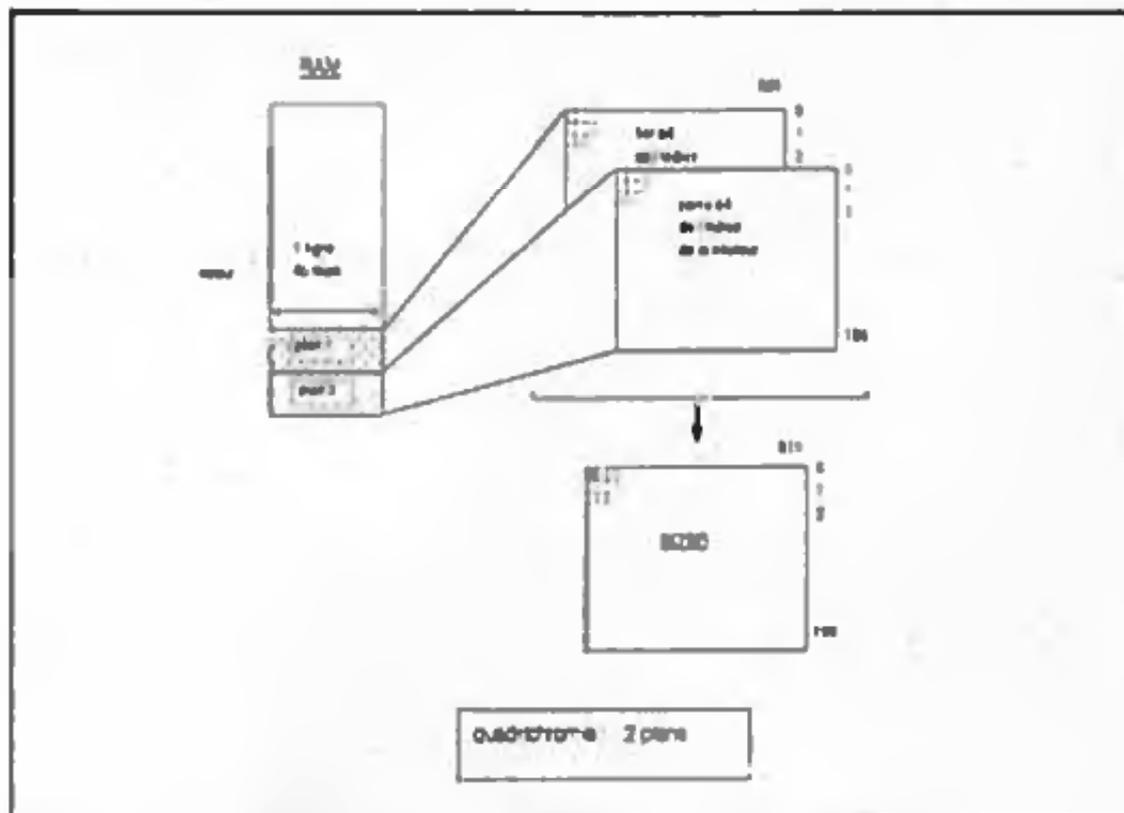
Même s'il dispose de milliers de tubes différents, le peintre n'emploie jamais que quelques couleurs à la fois. Pour plus de commodité, il les place sur sa palette. Il en va de même pour l'ATARI ST: il dispose de 512 couleurs différentes, mais n'en utilise au maximum que 16. En système binaire, il faut respectivement 9 et 4 bits pour représenter les nombres de 0 à 511 ou 15. La palette va profiter de cette différence de 5 bits pour économiser de la place pour l'affichage.

La palette est une suite de 16 mots (1 mot = 16 bits) contigus en mémoire qui identifient chacun une des 16 couleurs utilisables. Dans un mot, on place les proportions de rouge, de vert et de bleu qui composent une couleur. L'intensité de chaque teinte de base varie de 0 (absence) à 7 (maximale) et se code en trois bits. Il faut donc $3 \times 3 = 9$ bits pour la définir complètement. Lorsque l'on voudra employer une des 16(4) couleurs, il suffira d'indiquer, en quatre bits, sa place dans la palette (0...15). A cet endroit se trouvera sa description complète sur 9 bits. Si on n'utilise que quatre couleurs en même temps - en moyenne résolution - , on placera leur composition dans les quatre premiers mots de la palette. Il suffira alors de deux bits pour identifier une couleur parmi quatre : "00" binaire = 0 décimal , "01" = 1, "10" = 2, "11" = 3, qui désignent respectivement les couleurs dont la description se trouve dans le mot 0, 1, 2 ou 3 de la palette.



→ Le bitmap en 2 plans

En moyenne résolution, quatre couleurs distinctes sont possibles simultanément à l'écran. Une fois qu'elles sont choisies, leur composition en couleurs fondamentales (R + V + B - 3 bits pour chacune) est stockée dans les quatre premiers mots de la palette. Si l'on veut colorier un pixel dans la première couleur sélectionnable, il faut lui associer dans la mémoire graphique le nombre de 2 chiffres binaire "00". Chaque bit est stocké dans un plan séparé. Les deux plans se suivent en mémoire et occupent une zone de 32 Ko. Pour connaître la couleur d'un pixel, il faut donc aller chercher le bit dans le plan un, le deuxième dans le plan deux. On obtient alors un nombre de 0 à 3 qui pointe vers un des quatre premiers mots de la palette. C'est ce mot qui contient la description complète (9 bits) de la couleur du pixel.



⇒ *Basse résolution :*

320 x 200 pixels et 16 couleurs à choisir parmi 512. Les 16 mots de la palette sont utilisés. Il faut quatre bits pour désigner un de ces mots, donc la mémoire écran est divisée en 4 plans successifs.

Résumons-nous :

L'ATARI ST supporte deux types de moniteurs - monochrome et couleur - et trois résolutions: haute (640 x 400), moyenne (640 x 200) et basse (320 x 200). La haute résolution exige un moniteur monochrome. La moyenne et la basse résolution exigent un moniteur couleur. Elles permettent respectivement l'affichage simultané de 4 ou 16 couleurs choisies parmi 512.

2. LOGICIELS DE GRAPHISME

Une bonne machine ne vaut rien sans un bon programme. Cela paraît évident, et ça l'est en effet. Heureusement pour nous, les programmeurs se sont déchaînés pour l'ATARI ST et l'ont doté d'une gamme unique de logiciels graphiques.

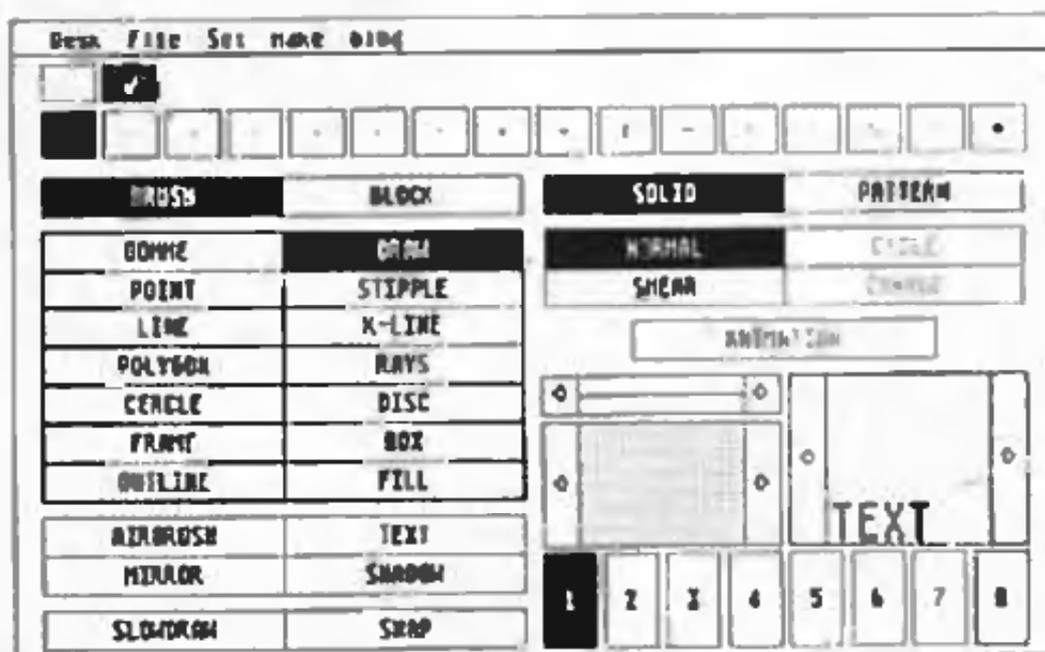
☛ D.A.O. orienté pixel

(Dessin Assisté par Ordinateur)

☛ DEGAS ELITE

Le grand classique

DEGAS ELITE, travaille alternativement avec deux écrans. Le premier est le tableau des commandes:



Ce sont les outils dont vous disposez. On choisit un outil en cliquant son nom avant de passer dans le second écran - la "feuille" de dessin. Les principaux outils permettent de:

- ✓ tracer des figures géométriques: ligne, cercle, rectangle, ellipse;
- ✓ peindre des zones dans une couleur ou un motif donné;
- ✓ écrire du texte;
- ✓ vaporiser des pixels (comme un aérographe).

Ce programme est facile à mettre en oeuvre, mais il a une limite: il manipule l'image comme il la stocke, c'est-à-dire dans un bitmap. Quand vous noircissez un pixel, il va donc basculer le bit correspondant dans le bitmap d'écran. Il en va de même pour toutes les fonctions. Une fois le dessin effectué, le programme ne sait plus comment il l'a construit. Il est donc incapable d'effacer un texte ou une figure géométrique quelconque. De plus, la taille et la résolution de l'image sont limités par l'écran. Stocker cette image revient à copier intégralement le contenu de l'écran - son bitmap - dans un fichier.

Degas Elite travaille dans les trois résolutions. La sortie de l'image se fait exclusivement sur imprimante matricielle ou laser.

➔ SPECTRUM 512

En gros, ce programme fait la même chose que DEGAS, mais lui, il le fait en 512 couleurs. Oui, vous

avez bien lu. Il peut afficher simultanément les 512 couleurs de l'ATARI ST dans une seule image.

➔ QUANTUM

En gros, ce programme fait la même chose que le précédent, mais il le fait en 4096 couleurs.

➤ D.A.O. orienté objet

C'est le domaine du dessin technique, des plans et des tracés.

➤ EASY DRAW :

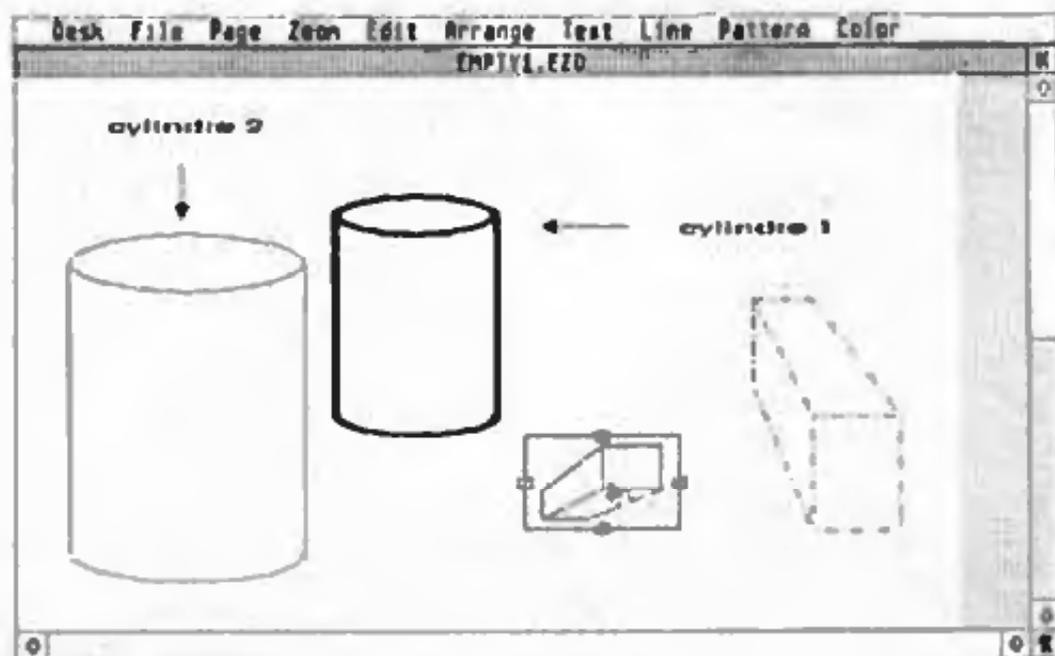
Ce logiciel a une approche toute différente du dessin.

1° : La surface de travail n'est plus l'écran, mais bien une feuille de format quelconque. Les ascenseurs permettent de faire "glisser" l'écran - la partie visible - sur toute la surface de la feuille. Son affichage peut de plus se faire à une échelle variable. Le mode "pleine page" permet d'avoir une vision globale du dessin final, même si la feuille qui le recevra est de format A3 ou A2. Le mode "zoom" autorise lui le positionnement précis des figures (points, lignes,...).

2° : Les objets géométriques (lignes, cercle, texte,...) apparaissant à l'écran sont "compris" du logiciel. Ils peuvent donc être déplacés, effacés, modifiés (agrandis, tournés, copiés, etc.) à volonté. Rien n'est jamais définitif avant l'impression.

3° : L'impression peut se faire sur une table traçante, une caméra ou une imprimante matricielle ou laser. Si ce dernier périphérique est choisi, le logiciel la pilotera en mode graphique au maximum de sa résolution. Quelle que soit l'échelle employée, une ligne sera toujours un ligne, et pas un escalier.

Donnons un exemple d'écran :



Les cylindres sont formés à partir d'une ellipse, d'un arc d'ellipse et de deux droites. Une fois dessinées individuellement, ces quatre figures sont regroupées logiquement pour former un seul objet : le cylindre. Le deuxième cylindre est obtenu par une duplication du premier suivie d'un agrandissement et d'un changement de l'épaisseur du trait. Il en est de même pour le parallélépipède rectangle.

➤ CAD 3D

C'est presque la même chose, sauf que c'est différent. On ajoute la troisième dimension - la profondeur -, des sources de lumière, des ombres et une caméra.

☛ Création d'un objet

Il y a trois manières de créer un objet en trois dimensions :

↳ 1° *faire tourner sa section autour d'un axe*

En faisant tourner un disque autour de son centre, on obtient une sphère.

↳ 2° *élever une surface*

En élevant un disque, on génère une colonne. On crée de la sorte buildings, planches, piliers, etc..

↳ 3° *combiner les objets*

Une fois les objets de base créés par une des deux , on peut les additionner ou les soustraire entre eux. En soustrayant d'un cylindre un autre cylindre plus étroit mais de même longueur, on crée un tube creux (à l'intérieur). En additionnant une planche plate et quatre piliers carrés, on crée une table. Cette technique est puissante, mais assez complexe à mettre en oeuvre.

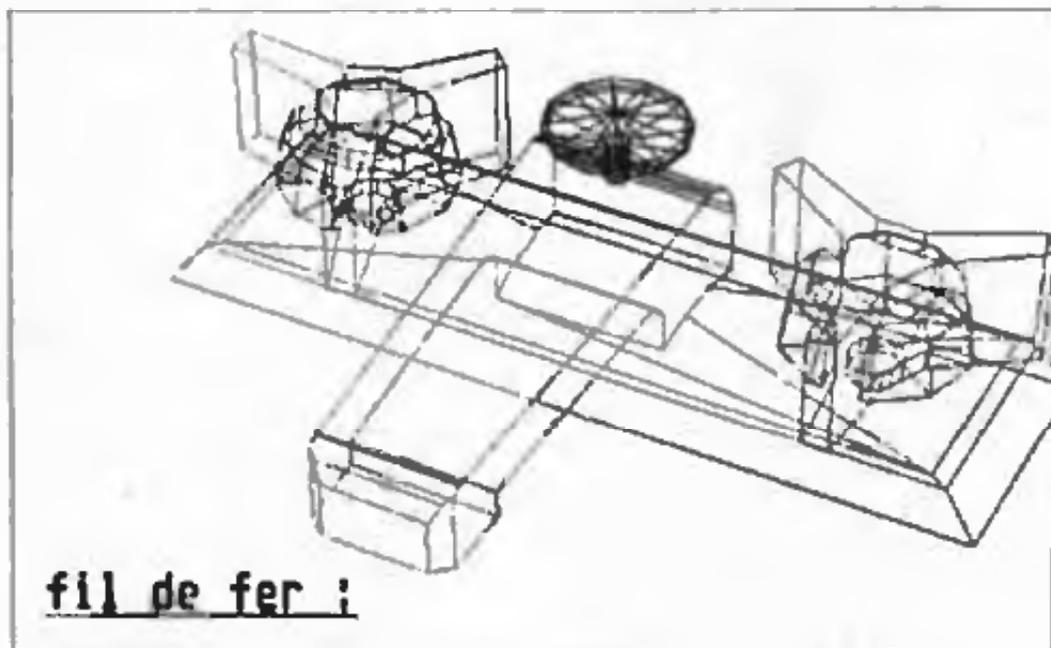
☛ Eclairage

Après la création d'un objet - une fusée, par exemple - vient l'étape d'éclairage, qui produira l'effet d'ombres. On éclaire un objet en disposant autour de lui des sources de lumière d'intensité variable.

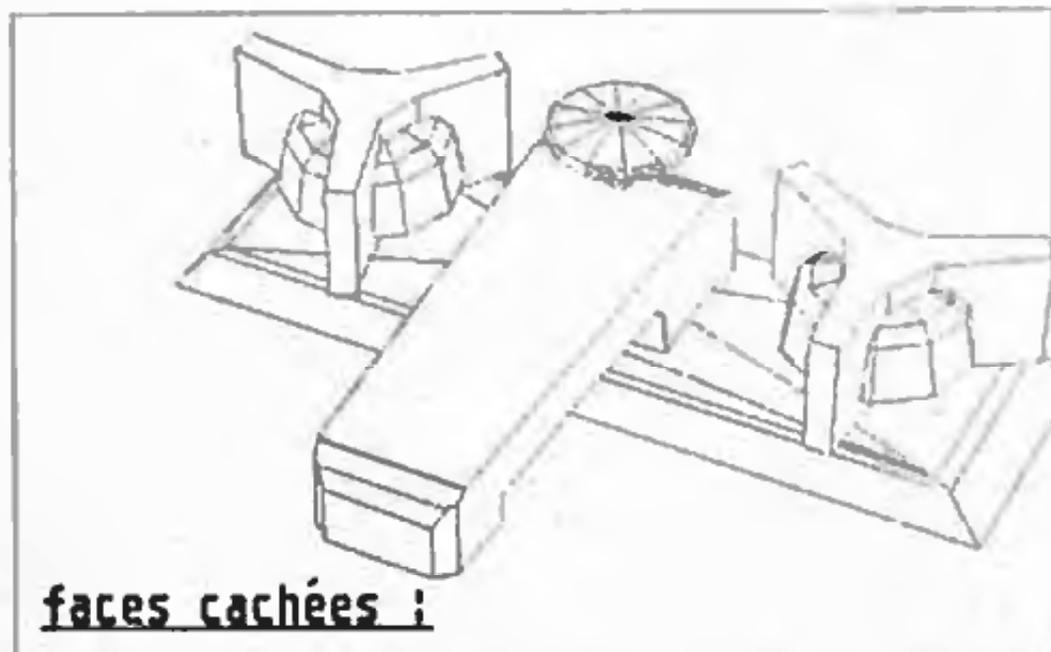
☛ Représentation d'une scène

Vient ensuite la visualisation. Les objets composant une scène peuvent être représentés de manière plus ou moins réaliste.

La représentation la plus simple est le mode "*fil-de-fer*" où seules ses arêtes sont visibles :

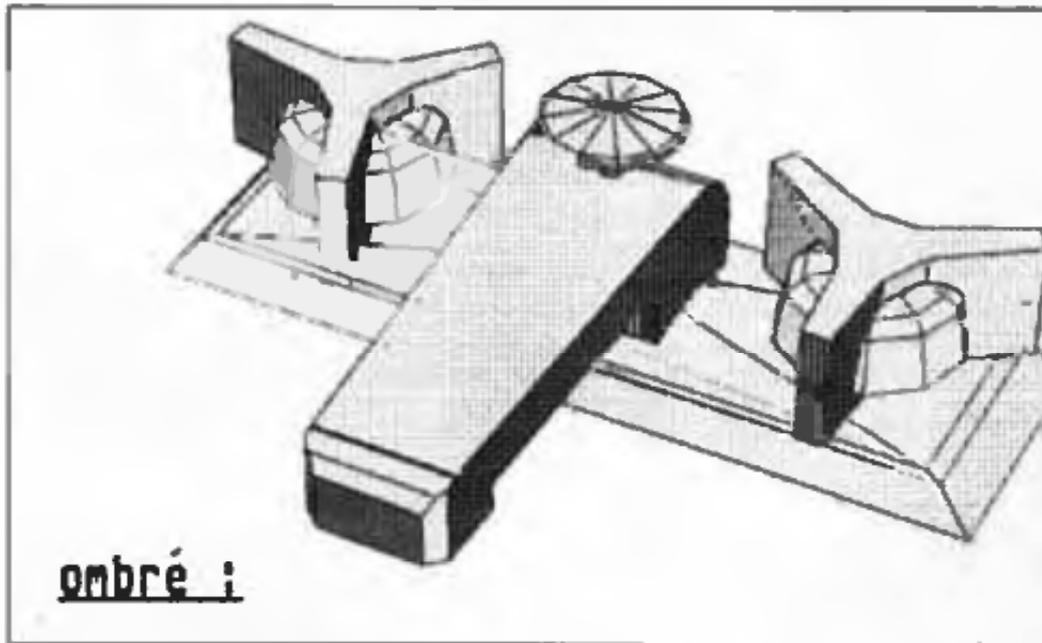


En mode "*faces cachées*", seules les parties visibles sont dessinées :



Plus le dessin est complexe et plus lent sera l'affichage.

Le troisième mode ajoute le traitement des ombres et éclairages.



☛ Déplacement

Les objets sont immobiles, mais on peut changer le point de vue. Pour cela, on déplace dans l'espace une caméra qui représente la position de notre œil. Si la caméra recule, l'objet s'éloigne. La perspective peut être renforcée ou affaiblie en rapprochant ou éloignant le point de fuite.

☛ Animation d'images

La quatrième dimension, le temps. Pourquoi s'en priver. Une fois votre fusée construite, regardez-la décoller et suivez-la.

↳ CYBER CONTROL

C'est un langage (Basic-like) d'animation qui pilote CAD-3D. Une fois votre fusée construite, il vous permet d'écrire un scénario: les déplacements des objets, les déplacements de caméras, les changements de lumière, etc.. Le défilement des images peut se faire en temps réel ou être enregistré en vidéo.

Oui mais, m'sieur, comment elle vole la fusée, je ne vois pas de flamme derrière ? Pas de problèmes ... voyez CYBER PAINT !

↳ CYBER PAINT

Il permet de retoucher chaque image et de parachever votre oeuvre.

A quand la sortie dans les salles ?

☛ C.A.O.

La Conception Assistée par Ordinateur

A côté des logiciels "classiques" de dessin technique existent des produits plus particulièrement destinés à une profession ou un usage particulier: architectes (*Arkey*), dessinateurs industriels (*Campus*), concepteurs de circuits électroniques (*Circuit Maker*)..

Ils ne se contentent pas seulement d'aider le professionnel à dessiner, mais ils l'assistent dans toutes les étapes de la conception. *Arkey*, par exemple,

répertorie les différents éléments (fenêtres, portes), calcule les métrages, prévoit les quantités de matériaux à utiliser, etc.

A côté de ces produits "sérieux" existe un logiciel étonnant: *Genesis*

➡ **GENESIS**

C.M.A.O.: Conception de Molécules Assistée par Ordinateur. Il permet de construire des molécules complexes à partir de leurs constituants (les atomes) et d'observer leur représentation spatiale. Les fonctions de visualisation permettent de voyager autour de la molécule, de la traverser, de mettre en évidence certaines structures.

☛ **Le Relief**

Mais oui, c'est possible. Beau, et pas cher (±150 \$). Avec les lunettes **STEREOTEK** (voir le chapitre consacré aux *PÉRIPHÉRIQUES* divers), l'image sort de votre écran. Elles s'utilisent directement avec tous les produits Antic (**CAD 3D**, **CYBER PAINT** et **GENESIS**).



**LES TRAITEMENTS
de TEXTES**



1. Introduction

Les programmes qui "traitent" du texte peuvent être répartis, suivant les fonctionnalités qu'ils offrent et les besoins auxquels ils répondent, en quatre classes ou types :

- ◆ 1. Editeur
- ◆ 2. Traitement de textes
- ◆ 3. Traitement de documents
- ◆ 4. P.A.O.

Les caractéristiques de chacun d'eux les destinent à des tâches et des utilisateurs différents. Avant de les examiner individuellement, nous allons définir certains termes usuels et faire quelques remarques communes à tous.

2. termes usuels

➤ CARACTERES ALPHABÉTIQUES

Les 26 caractères de l'alphabet plus les caractères accentués plus le "ç" + le "œ".

➤ CARACTERES ALPHANUMÉRIQUES

Les caractères alphabétiques plus les dix chiffres et l'espace.

➤ STYLES DE CARACTERES

Ce sont les cinq différentes façons d'afficher à l'écran et d'imprimer un caractère: gras, souligné, *Italique*, léger (grisé), exposant/indice. On peut les combiner entre eux de 32 (2⁵) manières différentes.

Remarque :

Le style léger est généralement réservé à l'écran car il rend très mal à

l'impression, sauf sur imprimante laser.

➤ **POLICE/FONTE DE CARACTERES**

Ensemble des caractères d'un même type caractérisés par leur dessin. Les plus courantes sont les polices SWISS, HELVETICA, DUTCH. Il existe aussi des polices de caractères non latins : arabe, hébreu, hindou.

➤ **TAILLE DES CARACTERES**

Exprimée en point (1/72^e de pouce), elle correspond à la hauteur de la cellule dans laquelle sont inscrits tous les caractères d'une fonte.

➤ **INTERLIGNE**

Distance en points séparant les cellules des caractères de deux lignes consécutives.

➤ **COMBINAISON DE TOUCHES**

C'est la pression d'une (ou plusieurs) touches spéciales - Control, Alternate, Shift - et d'une touche normale - toutes les autres - dans un ordre précis : d'abord enfoncer sans la relâcher la (les) touches spéciales et puis presser la touche normale.

➤ **WYSIWYG**

What You See Is What You Get (Ce que vous voyez est ce que vous obtenez).

L'origine de cet acronyme est un ancien show TV américain. Chaque semaine, pour terminer son émission, le présentateur apparaissait habillé en femme avant de lancer sa réplique : "What you see is what you get!"

Son usage dans la terminologie informatique est nettement moins ambigu. Et moins drôle aussi. On entend par WYSIWYG que l'image du document à l'écran est l'exacte représentation du résultat obtenu à l'impression. Sont donc fidèlement représentés le

dessin des caractères (fonte + style), leur taille, leur espacement, l'interlignage, etc.

Avant de continuer, coupons l'herbe sous le pied des coupeurs de cheveux en quatre: *What we see* sur l'écran n'est jamais tout à fait *what we get* à l'imprimante, pour la simple raison que ces deux périphériques n'ont pas la même résolution.

On peut distinguer plusieurs degrés dans le WYSIWYG:

1. Aucun.
2. Partiel.
3. Total.

⇒ 1. **AUCUN WYSIWIG:**

Aussi bizarre que cela puisse paraître, cela se trouve encore sur ATARI ST (*Textomat*).

On travaille à l'aveugle. Les différents styles (gras, soulignés...) sont matérialisés à l'écran par des codes de contrôles devant et derrière les mots qu'ils modifient. Cela rend de plus la justification impossible.

Une manière de pallier ce défaut est l'impression à l'écran: une fois le texte définitif, le programme envoie au moniteur l'image de ce que l'on obtiendrait à l'impression. *Défauts*: lenteur et impossibilité de modifier le document dans ce mode.

⇒ 2. **WYSIWIG partiel:**

C'est le cas de la majorité des programmes sur ST. Sont fidèlement représentés le style des caractères (gras, souligné...), la justification des mots et leur position dans la ligne. Les polices employées par l'imprimante ainsi que l'espacement proportionnel entre les caractères ne sont pas traduits à l'écran. Le *Rédacteur*, par exemple, autorise à l'impression

jusqu'à 19 polices proportionnelles différentes de taille variable (10 ou 12 points). Elles sont visualisées à l'écran par 19 autres polices, distinctes, non proportionnelles et de taille fixe.

Le multicolonnage, lorsqu'il est disponible, n'est pas toujours représenté à l'écran. Les colonnes sont alors créées séquentiellement et mises côte à côte lors de l'impression.

⇒ **3. WYSIWYG total:**

C'est le cas des processeurs de documents et des logiciels de PAO. Sont respectés à l'écran styles des caractères, dessin et fontes des tailles, espacements proportionnels et interlignage. Ajoutons que le moniteur monochrome s'impose dans ce cas.

Certains se demandent peut-être pourquoi cet affichage performant n'est pas disponible pour les traitements de textes ou les éditeurs. Après tout, nous l'avons vu au chapitre GEM: le système offre au programmeur tous les outils nécessaires pour travailler en WYSIWYG total. C'est vrai, mais, ici comme ailleurs, tout se paye, même le WYSIWYG. L'affichage simultané d'images et de caractères en plusieurs styles, tailles et polices différentes prend du temps, beaucoup de temps. Et ce qui est acceptable pour un programme de mise en page ne l'est plus pour un programme dont la tâche première est de faire de la saisie rapide de texte.

3. Remarques communes

↳ FICHIERS MULTIPLES

Les programmes les plus élémentaires ne permettent de travailler que sur un fichier à la fois. Pour en manipuler un autre, il faut alors 1° sauver le fichier courant et 2° charger le nouveau. C'est un processus assez lent.

Il est parfois très utile de voir un second document pendant qu'on travaille. C'est pour cela que de nombreux logiciels autorisent la présence simultanée en mémoire de plusieurs fichiers. A chacun d'eux est associé, dans l'environnement GEM, une fenêtre que l'on dimensionne et positionne à souhait. Le transfert d'information de l'une à l'autre ne peut se faire que si le logiciel dispose d'un presse-papier.

↳ RAPIDITÉ

Elle est synonyme de confort et s'applique principalement dans trois cas bien précis : l'affichage, la sauvegarde et les manipulations de textes.

→ Rapidité de l'affichage

Nous avons vu que les routines de GEM s'occupant du scrolling (déroulement) ne sont pas très performantes. C'est pour pallier cette lacune que plusieurs concepteurs ont implanté dans leur logiciel de nouvelles routines de défilement de l'écran. C'est le cas, entre autres, de l'éditeur **TEMPUS** et du traitement de texte *Le Rédacteur*. Le résultat est époustouflant: le redessin d'une page complète est presque instantané.

Un travail soutenu exige un moniteur monochrome car l'affichage couleur fatigue plus les yeux. La cause en est d'une part le rafraîchissement moins fréquent de l'écran couleur qui entraîne un certain clignotement de l'image et, d'autre part, la matrice

plus grossière dans laquelle sont dessinés les caractères: 8 x 8 pixels au lieu de 16 x 8 en monochrome. Ils sont donc moins faciles à distinguer.

→ **Rapidité des manipulations de texte**

Aucun logiciel n'est égal devant le chronomètre. Et vous seriez surpris : du meilleur au pire des cas, vous devrez attendre - gaiement - de quelques secondes à plusieurs minutes, pour exécuter une même fonction. Ayons pitié. Ne nommons pas le traitement de texte qui demandait plus d'une heure pour recopier dix pages d'un endroit à un autre (authentique). Il est des plaisirs dont on aimerait pouvoir se passer.

→ **Rapidité de la sauvegarde/chargement**

Certains logiciels (SIGNUM, par exemple) compriment les documents avant de les stocker sur disquette. Cela a un triple avantage : occupation moindre de disquette et accélération du chargement et de la sauvegarde du document.

Après ces quelques notions, examinons les quatre classes de logiciels de la manière suivante:

- ⇒ Le quoi : ce qu'il font et comment il le font.
- ⇒ Les fonctions disponibles.

Conventions :

✓ Les fonctions absolument nécessaires et que l'on retrouve (presque) partout sont précédées d'un point.

Exemple : .bloc

Si une d'elles venait à manquer, mieux vaut pour vous chercher ailleurs.

✓ Celle qui sont très pratiques - voire nécessaires - mais dont la disponibilité n'est pas systématique sont précédées d'un "+".

Exemple: + Undo

✓ Les fonctions raffinées moins courantes sont signalées par un "!".

Exemple: ! déroulement parallèle

4. Les quatre types de programmes

Editeur :

☛ Quoi ?

C'est le plus simple de tous. Il permet de mettre au point les textes des programmes sources. Le but est ici de produire un fichier contenant du texte ASCII pur, utilisable directement par un compilateur. On ne dispose donc pas d'effet de style, de mise en page ou de justification. Les lignes sont le plus souvent limitée à 80 caractères.

☛ Pour qui ?

Les programmeurs exclusivement.

☛ Fonctions

▼ *Fonctions de base :*

Effacement, copie, déplacement de caractère, mot ou ligne.

▼ • *Manipulation de bloc :*

Un bloc est une partie du document qui est considérée comme une seule unité et que l'on peut copier, effacer ou déplacer en une fois. On le délimite à l'aide de la souris et sa prise en compte est visualisée par un changement d'aspect à l'écran (vidéo inverse ou grisé).

▼ + *undo*

L'absolutoire. Elle vous pardonne votre dernier péché (effacement de texte). Si cette fonction est disponible dans un programme, elle est automatiquement associée à la touche prédéfinie [UNDO]. C'est aussi une manière facile de déplacer ou copier du texte d'un endroit vers un autre: 1° on l'efface et 2° on le récupère par UNDO où on le désire.

▼ + *fichiers multiples*

Permet de décomposer le code source en plusieurs modules, accessibles simultanément. N'est utile que si le langage autorise la programmation en modulaire.

▼ + *manipulation de fichiers*

Si votre disquette ne contient plus assez de place pour recopier le fichier de travail, il est absolument nécessaire de pouvoir en formater une nouvelle ou bien de détruire des fichiers devenus inutiles, sans quitter votre programme principal. Il doit pour cela offrir les fonctions *ad hoc*. Elles sont souvent regroupées dans le menu "Disque".

Traitement de textes

• Quoi ?

Le but visé ici n'est plus la production d'un fichier (comme l'éditeur), mais bien la saisie d'un texte et son impression. Ils permettent la rédaction de lettres ou de rapports de présentation simple.

• Fonctions

▼ • *Frappe kilométrique*

Les caractères sont saisis en continu, le programme insérant lui-même les retours de chariot.

▼ • *Justification automatique*

Des espaces de longueurs variables sont insérés entre les mots, de façon à obtenir un texte cadré à la fois sur les marges gauche et droite. Le blanc insécable permet d'éviter que deux mots ne soient séparés par un retour chariot. Cela permet d'écrire, par exemple, "ad hoc" correctement. La justification automatique peut dans certains cas être aberrante. Si le mot *anticonstitutionnellement* dépasse d'un caractère la longueur de la ligne, il passe à la suivante et 26 blancs sont insérés dans cette ligne-ci. La césure automatique est un algorithme intégré au programme qui coupe en deux un mot trop long pour éviter cette situation.

▼ • *Feuille de style (layout)*

Elle définit la forme générale du document : nombre de lignes par page, texte de pied et d'en-tête et parfois d'autres attributs tels que espacement des lignes, type de caractères (élargis, condensés), etc.

Elle est créée avant le document, et elle s'applique automatiquement à toutes les pages de celui-ci. La

plupart des logiciels n'acceptent qu'une feuille de style par document.

▼ + *Gestion de notes*

L'attribution d'un numéro de référence et l'impression en bas de page sont gérées par le programme.

▼ + *Presse-papier (Clipboard)*

C'est une amélioration du principe des blocs permettant de les transférer ou copier d'un document à un autre. Le stockage d'un texte dans le presse papier écrase son contenu précédent. Bien que d'une grande utilité, elle est absente de bien des logiciels.

▼ + *Glossaire (abréviations)*

Il contient des séquences de caractères (mot, phrase, paragraphe) que l'on peut rappeler d'une simple combinaison de touches. Certains logiciels permettent de sauver, en plus du texte, sa mise en page et le style des caractères.

▼ + *Raccourci équivalent-clavier*

La grande majorité des logiciels emploient l'interface GEM. Si c'est un gage évident de facilité pour l'utilisateur, il en résulte parfois une certaine lourdeur d'utilisation. En effet, l'obligation de quitter trop fréquemment le clavier pour manipuler la souris provoque une baisse de rendement et une perte de confort non négligeables. Une solution existe: le "raccourci-clavier". C'est l'association d'une commande manuelle (au clavier) à une fonction particulière du programme. On peut donc l'exécuter soit à partir du menu, soit grâce à une combinaison de touches (keystroke) au clavier.

L'apprentissage des raccourcis-clavier est facilité par leur choix mnémorique. Il s'agit souvent de la première lettre du nom de la fonction : Sauve-

garde, Quitter le programme, etc. La combinaison de touches peut aussi être rappelée dans le menu, à côté de l'action qu'elle exécute.

▼ ! *Déroulement parallèle*

Lorsque plusieurs fichiers de travail sont visibles simultanément à l'écran, leur déroulement peut être relié. Tout changement de ligne ou de page écran du curseur est automatiquement répercuté dans les autres fenêtres. C'est une fonction très utile pour les traducteurs : le texte original occupe une moitié de l'écran, et sa traduction l'autre.

▼ ! *Publi-postage*

Permet l'impression de documents personnalisés. Les rubriques variables (nom, adresse, code postal) en provenance d'un fichier externe sont insérés dans une lettre type qui est imprimée plusieurs fois en changeant seulement les données variables.

▼ + *Chargement/sauvegarde de fichier*

Un traitement de textes ajoute aux caractères tapés par l'utilisateur divers codes de contrôle. Cela permet de stocker aussi bien le texte que le style des caractères et la mise en page du document.

Chaque logiciel, ou presque, utilise ces codes de contrôle d'une manière différente. Si vous voulez récupérer ou échanger des textes construits avec un autre programme, il faudra alors employer des conventions communes. Beaucoup de logiciels permettent de sauver ou charger des documents dans le format *1st Word*. En l'absence de format commun, il est toujours possible de sauver le document en ASCII. Il est disponible sur tout les traitements de texte, mais il ne permet de récupérer que le texte. Le style des caractères et la mise en page sont alors perdus.

▼ ! *Gestion index / table des matières*

La gestion d'index consiste à créer un fichier séparé du texte courant où l'on place une suite de mots choisis et les numéros de pages où ils se trouvent. La gestion de table des matières reprend les titres et numérotation des chapitres, paragraphes et autres.

▼ ! *Correcteur orthographique*

Le rêve des potaches. Les bons sont rares. Patience. Attention. Certains logiciels (*Beckertext* et *1st Word Plus* e.a.) sont vendus avec un pseudo-correcteur orthographique. Ils ne s'agit en réalité que d'une liste de mots connus par le programme, que l'on peut enrichir de ses propres termes. Le "correcteur" se borne à signaler les mots de votre texte qui ne sont pas contenus dans sa liste. Il faut donc y placer séparément pluriel et singulier, féminin et masculin ainsi que les conjugaisons complètes (temps et personne) des verbes. Même énorme, la liste ne peut jamais contenir qu'une partie restreinte du vocabulaire courant.

▼ ! *Dictionnaire des synonymes (thesaurus)*

Disponibles dans les logiciels anglophones (*Protext* e.a.), ils sont une aubaine pour l'écrivain en panne. La francisation se fait attendre.

▼ + *Impression en arrière-plan (Spooling)*

Elle permet de continuer à travailler sur un document pendant qu'un autre s'imprime.

Traitement de documents :

• Quoi ?

Cette dénomination peu courante désigne une classe de programmes assez récente, faisant la transition entre les traitements de textes classiques et les logiciels de P.A.O. Ils conviennent parfaitement à l'édition de textes scientifiques. *SIGNUM* et *WordUp* sont deux exemples de traitement de documents.

• Fonctions

▼ Fonctions de création

Contrairement aux logiciels de P.A.O., ils possèdent la plupart des fonctionnalités des traitements de textes.

▼ Polices de caractères

Un même document peut faire appel à plusieurs polices de caractères, proportionnelles, en plusieurs styles et tailles différents. On peut disposer de tous les symboles mathématiques, des lettres grecques, etc.

▼ Interlignage/Kerning

On peut régler individuellement l'espace entre deux lignes (interlignage) ou caractères (successifs).

▼ Images

Elles peuvent être importées comme du texte et placées n'importe où.

▼ WYSIWYG total

Le document apparaît comme il sera imprimé et en particulier l'espacement proportionnel entre les caractères.

P.A.O. :

• **Fonctions**

Par rapport au traitement de documents, on rencontre les différences suivantes :

▼ *Importation du document*

Il est nécessaire de préparer le texte à mettre en page avec un traitement de texte séparé avant de l'importer dans le programme de P.A.O. Celui-ci n'est en effet pas prévu pour la saisie de grande quantité de texte.

▼ *Arrangement du texte*

Il peut être disposé en plusieurs colonnes, comme dans les journaux.

▼ *Graphisme*

Les images peuvent être importées de l'extérieur ou créées au sein même du logiciel. Il est aussi possible de changer leur taille (Zoom avant ou arrière) ou de n'en prendre qu'une partie.

▼ *Contour d'image*

Lorsqu'une image recouvre partiellement un texte, on peut demander à celui-ci de la contourner.

Ces programmes permettent de réaliser très facilement des rapports ou documents d'aspect sophistiqué. Ils manipulent aisément de grandes quantités d'informations permettant de réaliser livres ou manuels. La taille du document imprimé peut varier de la carte de visite au format A4. Bien qu'ils supportent les imprimantes matricielles (9 ou 24 aiguilles), ces

logiciels nécessitent une imprimante laser pour profiter pleinement de toutes leurs capacités. Certains (Publishing Partner, par exemple) sont même capables de piloter une Linotype, photocomposeuse professionnelle dont la résolution d'impression peut atteindre plus de 2.500 dpi - contre 300 pour les imprimantes laser- (voir le chapitre consacré aux *PÉRI-PHÉRIQUES*).

**LA MUSIQUE
ET LA NORME
MIDI**

Circuit sonore :

La manière la plus simple de produire un son est de programmer le processeur sonore - le YM-2149 - à partir d'un langage quelconque. En BASIC ST, on dispose à cette fin de deux instructions: *Sound* et *Wave*.

→ **SOUND** envoie une note sur un des trois canaux en précisant son volume ainsi que sa durée.

→ **WAVE** autorise la programmation polyphonique et le choix de l'enveloppe.

Le son est généré par le haut-parleur du moniteur ou de la télévision.

Son Digitalisé :

Il existe une manière simple de rendre un logiciel bavard: le son digitalisé. La première étape consiste à numériser le son (voix, musique, bruit), à le reproduire et à le stocker dans un fichier. C'est le rôle du digitaliseur sonore. Dans un deuxième temps, le son digitalisé est restitué, soit par le haut-parleur du moniteur, soit par un synthétiseur sonore.

Le digitaliseur et le synthétiseur sont souvent réunis dans un même boîtier que l'on connecte sur le port cartouche.

M.I.D.I. :

La plupart des possesseurs d'ATARI ST savent que l'interface MIDI est liée à la musique. Mais pourquoi,

et comment ? "That is the question", comme dit ma concierge.

Avant tout, rassurons les puristes : le ST est là pour traiter des informations, et pas pour faire de la musique. Il n'est pas musicien, mais seulement chef d'orchestre.

➤ *Instruments électroniques*

La configuration minimale pour "faire" de la musique électronique est composée d'un clavier et d'un *expander* (module de génération sonore). Comme le ST, ils sont pourvus de prises MIDI. Relions-les par un câble partant de la prise MIDI OUT du clavier vers la prise MIDI IN du module (note : un *synthétiseur* comprend le clavier et l'*expander* dans le même boîtier).

➤ *Les événements MIDI*

Le clavier ne produit aucun son, ce n'est pas son rôle. Si vous enfoncez une de ses touches, il envoie vers l'*expander* une suite de nombres qui déterminent le son à produire. Le premier indique l'enfoncement d'une touche ("NOTE ON") et le deuxième précise la position de cette touche. Lorsque vous relâchez la touche, un message "NOTE OFF" ainsi que le code de la touche suivent le même chemin. Vous venez de rencontrer vos deux premiers événements MIDI.

Passons à l'étape suivante : on intercale le ST entre le clavier et l'*expander*. Les messages vont donc transiter par l'ordinateur.

➤ *Le séquenceur*

Ce programme joue un peu le même rôle que les enregistreurs multipistes : il enregistre les événements MIDI qui traversent le ST et les sauve dans un fichier. Chaque instrument correspond à une piste (les séquenceurs les plus performants travaillent sur 64

pistes). Un *séquenceur* permet en plus de modifier le contenu d'une piste (changer des notes par exemple), de la décaler dans le temps, et de faire un tas d'autres choses irréalisables avec un enregistreur classique.

➤ ***L'éditeur de son***

L'expander est un appareil souple : en plus des différents sons prédéfinis (flûte, piano, biniou), il permet au musicien de créer ses propres sons. Cette création se fait avec un éditeur de son. Ce programme vous aide à construire, paramètre par paramètre, un nouveau type de son. Une fois au point, il ne vous reste plus qu'à l'envoyer, via le câble MIDI, dans la mémoire de l'expander.

➤ ***L'éditeur de partition***

Il ne faut plus connaître le solfège pour écrire la musique. Il vous suffit d'enregistrer sur le *séquenceur* votre oeuvre impérissable puis de la faire traduire en la partition équivalente et, enfin, d'imprimer par l'éditeur de partition. C'est le cauchemar des copistes professionnels. Les petits métiers foutent le camp.

➤ ***Les instruments MIDI***

Il existe, en plus du clavier, une large gamme d'instruments (guitare, batterie, harmonica, accordéon, saxo...) qui, au lieu de produire directement un son, génèrent un événement MIDI.

➤ ***La norme MIDI***

Lorsque les premiers instruments électroniques firent leur apparition, chaque constructeur avait sa propre norme et n'était compatible qu'avec lui-même. Il était donc impossible de relier des appareils d'origines diverses. C'est pour remédier à cette situation que la norme MIDI fut développée. Elle énonce les spécifici-

tés physiques (type de liaison, vitesse de transfert) et logiques (les messages) à respecter. La communication entre les instruments MIDI se fait par une liaison série (comme le RS-232) à 31250 bips (bits par seconde).

Et les jeux alors ?

Oui, et les jeux. Il faut le dire, que "Karate Kid II" et "Marble Madness" produisent du son MIDI.

Ah oui, j'allais oublier: MIDI veut dire *Musical Instrument Digital Interface* (Interface digital pour instrument de musique).

Si vous désirez en savoir plus, un autre volume de cette collection est entièrement consacré à la musique MIDI sur ordinateurs (*L'Indispensable pour la Musique assistée par ordinateur, M.S. N° 862*).

LES PÉRIPHÉRIQUES

Ils seront abordés dans l'ordre suivant :

- ⇒ 1° Périphériques de stockage.
- ⇒ 2° Périphériques d'entrée.
- ⇒ 3° Périphériques de sortie.
- ⇒ 4° Extensions.
- ⇒ 5° Périphériques divers.

1. Périphériques de stockage

-
- Lecteur de disquettes
 - Disque dur
 - FD-10
 - CD-ROM
 - Solid State Disk
-

➤ Lecteur de disquettes

Le lecteur de disquettes - ou Floppy Disk Drive - est sans conteste le moyen le plus répandu de stockage de données. Plus lent et de moindre capacité qu'un disque dur, il possède sur celui-ci un avantage non négligeable: il utilise des disquettes amovibles. Elles

sont le moyen idéal pour transporter, échanger et conserver des données.

On peut exploiter sur ATARI ST plusieurs tailles - 3'1/2 et 5'1/4 - ainsi que plusieurs formats de disquettes.

⇒ 3'1/2

Le lecteur de disquette 3'1/2 fait partie de la configuration minimum de l'ATARI ST. Il en existe deux modèles : simple face et double face. Tous deux utilisent un même format : 80 pistes par face, et 9 secteurs par piste.

Le modèle *simple face*, qui tend à disparaître, existe en deux présentations : externe et interne. La version externe peut s'utiliser avec tous les ST. La version interne équipe uniquement les anciens modèles de 520 STFM.

De la même manière, on trouve deux sortes de lecteurs *double face*. Le modèle interne équipe les 1040 et Méga ST, ainsi que les nouveaux 520 STFM. Le modèle externe peut s'utiliser avec tous les ST.

D'autres marques qu'ATARI proposent des lecteurs 3'1/2. Ils possèdent des caractéristiques identiques et peuvent donc s'utiliser sans aucun problème.

Rappelons pour terminer qu'un lecteur double face peut lire une disquette simple face, mais que le contraire n'est pas vrai.

⇒ 5'1/4

Deux raisons peuvent vous pousser à acheter un lecteur de disquettes au format 5'1/4: l'économie ou l'échange de fichiers MS-DOS. Quelle que soit votre raison, surveillez le nombre de pistes lues.

→ *Economie*

Les disquettes 5'1/4 sont beaucoup moins chères que les 3'1/2. Elles sont donc le support idéal pour conserver à peu de frais les copies de sécurité - back-up - de vos disquettes 3'1/2 ou de votre disque dur.

→ *Fichiers MS-DOS*

Nous avons vu que le TOS utilisait le même format de stockage de données sur disquette que MS-DOS, le système d'exploitation des IBM et compatibles. Il reste cependant un obstacle au libre échange de données : la taille. Bien qu'IBM se soit rallié au 3'1/2 pour ses nouveaux ordinateurs, la majorité des PC existant sont équipés exclusivement de lecteurs 5'1/4. Il faut donc utiliser ces disquettes pour l'échange de données.

Il existe un lecteur 5'1/4 dans la gamme ATARI (le PCF 554) mais aussi dans d'autres marques.

→ *Nombre de pistes*

Tous les lecteurs de disquettes 5'1/4 ne sont pas égaux entre eux. Ils peuvent différer par le nombre de pistes qu'ils savent lire. Le ST, nous l'avons vu, utilise 80 pistes par face. La majorité des PC en emploient 40. Il faut savoir qu'un lecteur 5'1/4 80 pistes ne sait pas forcément lire une disquette formatée en 40 pistes. On trouve donc 3 types de lecteurs 5'1/4 sur le marché : 40, 80 et 40/80 pistes.

Notons pour terminer qu'un lecteur double face 40 pistes, comme le PCF 554, ne permet de stocker sur une disquette que 360 Ko, ce qui le rend moins intéressant si vous l'utilisez pour faire des back-up. Le formatage devra se faire avec un programme spécial, car le TOS ne le fait qu'en 80 pistes. Un lecteur mixte - 40/80 - est donc beaucoup plus avantageux.

☛ Disque dur

Il est des moments où la lenteur relative du lecteur de disquettes et sa faible capacité de stockage deviennent inconfortables. Pour ces deux problèmes, le disque dur offre une solution.

→ *Vitesse*

Le moteur qui anime le disque dur tourne constamment, à une vitesse bien plus élevée que dans le lecteur de disquettes. Cela a deux conséquences: on accède plus vite aux données - un tour de piste est plus rapide - et elles sont plus vite transférées. La vitesse de transfert est encore accélérée par le raccord direct du disque dur au bus DMA de l'ATARI.

→ *Capacité*

La rigidité du disque dur permet d'employer une technologie plus pointue : la tête de lecture est plus fine et plus proche de la surface magnétique que dans un floppy drive. Cela permet de stocker plus d'informations sur une même surface.

→ *Partitions*

On trouve sur le marché une large gamme de disques durs, d'une capacité variant de 10 à 195 Mo. Le TOS, comme MS-DOS, ne peut cependant pas gérer de bloc de données plus gros que 16 Mo. Il faut donc diviser logiquement le disque dur en plusieurs partitions, de sorte qu'aucune ne dépasse 16 Mo. Cette opération se fait lors du formatage initial du disque dur.

→ *Chaînage (daisy-chaining)*

Bien que l'ATARI ST ne dispose que d'une sortie pour connecter un disque dur, on peut en utiliser jusqu'à

huit simultanément. Chacun est muni pour cela de deux ports de connexion: **DMA input** et **DMA output**. On procède par chaînage, exactement comme pour les lecteurs de disquettes (voir le chapitre des **GÉNÉRALITÉS**). Remarque importante : le disque dur **ATARI** ne possède pas de sortie DMA. Il doit donc travailler seul ou en fin de chaîne.

→ *les modèles*

Ils se distinguent entre eux par leur capacité - en Mo - et leur vitesse - en micro-seconde (ms). Dans une même gamme, les plus gros sont souvent plus rapides. Certains modèles présentent des caractéristiques supplémentaires intéressantes, telles qu'une horloge permanente - comme sur le Méga ST -, un port SCSI ou un lecteur de disquettes - 3'1/2 ou 5'1/4 - intégré dans le même boîtier.

☛ **FD-10**

Le chaînon manquant.

Depuis plus de quinze ans, deux types bien distincts de supports magnétiques se partagent, sans heurts, le marché. D'un côté, les *mous* : amovibles, lents, peu chers et de faible capacité. De l'autre, les *durs* : fixes, rapides, plus chers et de grande capacité. Un produit d'un genre nouveau jette désormais un pont entre ces deux mondes : le FD-10, de Supra. Comme son nom peut le laisser deviner, il s'agit d'un ***Floppy Disk drive de 10 Mo.***

☐ Caractéristiques:

⇒ Support

Il utilise des disquettes 5'1/4 à très haute densité: 480 *TPI* (Tracks Per Inch - pistes par pouce).

⇒ Capacité

Une fois formatée, la disquette laisse à l'utilisateur plus de 10Mo, soit l'équivalent d'une trentaine de disquettes simple face.

⇒ Vitesse de transfert d'informations

Entre celles du floppy drive et du disque dur. Il est connecté au ST par le port DMA.

✓ **Port DMA output** : permet le chaînage;

✓ **Horloge permanente** : préserve l'heure;

✓ **Format IBM** : il peut lire les disquettes 5'1/4 au format 360 Ko ou 1.2 M des IBM-PC. Il reconnaît automatiquement le type de floppy inséré et l'exploite correctement.

□ Avantages:

Il offre donc la mobilité d'une disquette, la capacité d'un petit disque dur pour une vitesse de transfert intermédiaire. Cela fait de lui le moyen idéal pour déplacer, échanger ou stocker de grandes quantités d'informations ainsi que pour faire des back-up de disque dur.

□ Désavantages:

Il faut cependant citer un inconvénient de poids : son prix. Les technologies récentes sont toujours plus chères. Au coût d'utilisation, s'ajoute l'achat initial du lecteur, le tout dépassant le prix d'un disque dur de 30 ou 40 Mo. Il ne conviendra donc pas à tous.

☛ CD-ROM

Le CD-ROM est un périphérique de stockage qui utilise comme support le disque compact (CD) et n'autorise que la lecture d'information (ROM: Read Only Memory).

Basé sur la même technologie que le CD audio - même support, même lecteur laser -, le CD-ROM en diffère par deux points. Le premier est le procédé de correction d'erreurs. Si nos oreilles sont relativement tolérantes, tel n'est pas le cas d'un ordinateur. Il suffit qu'un bit change de valeur pour que l'information perde tout son sens. Cela a donc nécessité que l'on mette au point un système beaucoup plus performant que la correction audio. La seconde différence est la sortie du résultat. Un CD-audio doit produire un son analogique. Il dispose pour cela d'un convertisseur numérique analogique. Un CD-ROM, par contre, doit envoyer le résultat sous sa forme numérique. Le convertisseur est donc inutile.

□ Le CDAR-504

Annoncé en 86, montré en 87 et produit fin 88, le CDAR-504 d'ATARI a su se faire attendre. Il est à la fois CD-ROM et CD-audio (d'où le "A" dans son nom). La mécanique de base étant la même, il a suffi d'ajouter un convertisseur N/A à un CD-ROM, ou si vous préférez, d'améliorer le système de correction d'erreur d'un CD-audio.

Contentons-nous d'examiner ses caractéristiques informatiques :

✓ Il lit des disques compacts. Chacun d'eux peut contenir 550 Mo.

✓ Comme pour le disque dur, le transfert d'informations vers l'ordinateur se fait via le port DMA, ce qui lui assure une vitesse maximum. Il possède une deuxième broche DMA qui lui permet d'être chaîné aux autres périphériques DMA.

□ Le standard CD-ROM

Pour éviter une anarchie analogue à celle créée par la diversité des formats de lecteurs de disquettes, les principaux fabricants de CD-ROM - Philips et Sony - se sont réunis en 1985. Ils ont établi, sous le nom de "*Yellow Book*", une liste de spécifications à respecter qui définit un format unique. Grâce à cela, tout lecteur CD-ROM, de quelque marque ou modèle qu'il soit, peut lire tout CD, quelle que soit son origine.

□ La standardisation des fichiers

"Tout programme peut exploiter tout fichier." Elle commence doucement à s'installer. Jusqu'à présent, presque toutes les applications spécialisées - encyclopédie, base de données financières, astronomiques, - utilisent des données stockées suivant un format qui leur est propre. Il faut donc un logiciel de recherche d'information par application. Le nouveau standard permettra à tout logiciel de recherche d'exploiter les informations présentes sur n'importe quel disque compact.

□ Usage

Le CD-ROM ne permettant pas l'écriture d'information, son emploi le destine à la conservation d'énormes masses de données statiques : dictionnaires,

encyclopédies, bases de données juridiques, astronomiques, etc.

□ **Son**

Les informations stockées peuvent être, naturellement, du texte, des images, mais aussi du son. Il est restitué par un synthétiseur sonore externe ou par le haut-parleur du moniteur. Cette dernière solution donne cependant un piètre résultat. L'information sonore peut contenir musique, bruit et parole. Cela permet d'accompagner texte et image d'un commentaire, en une ou plusieurs langues.

☛ Solid State Disk

C'est un périphérique d'un genre nouveau qui se branche sur le port cartouche de l'ATARI ST. Il comporte 1Mo de puces RAM - comme la mémoire de votre ordinateur -, une alimentation séparée et une batterie de sauvegarde.

☐ Utilisation

Son espace mémoire peut être géré comme un lecteur de disquettes supplémentaire, et/ou comme un *spooler* d'imprimante. Il s'utilise au même titre qu'un lecteur classique. Pour vous, il ressemble à un disque dur ultra-rapide de faible capacité. Sa nature différente est totalement transparente à l'utilisateur.

☐ Avantages

✓ *Rapidité* : la lecture ou l'écriture d'information sur ce "disque" se fait cinq fois plus vite qu'avec un disque dur.

✓ *Permanence de l'information* : sa batterie lui permet de conserver les données lorsque vous éteignez l'ordinateur.

Il existe actuellement en deux versions : le MEGADISK (1Mo + batterie de sauvegarde) et le POLYDISK (520 Ko, pas de batterie).

2. Périphériques d'entrée

- **clavier**
- **souris**
- **joystick**
- **rackball**
- **scanner**
- **tablette graphique**
- **digitaliseur vidéo**
- **digitaliseur sonore**

➤ Clavier

Tous les modèles de la gamme ST sont fournis avec un clavier standard, détachable ou intégré. Bien que fourni exclusivement avec les Méga ST, le clavier détachable peut aussi être utilisé sur les 520 ou 1040. Il suffit pour cela de le connecter, à l'intérieur de l'unité centrale, à la place du clavier intégré que l'on doit donc débrancher.

Il est également possible d'employer des claviers de type IBM, mais cela nécessite une interface matérielle adaptée.

☛ **Souris**

La souris mécanique ATARI est, pour l'instant, le seul modèle disponible. Elle demande, pour parcourir l'écran, une surface minimale de 8 x 13 cm, ce qui correspond à une résolution de ± 150 dpi.

En cas de panne, vous pouvez utiliser le clavier comme souris de secours. Les déplacements du pointeur à l'écran peuvent être obtenus par la pression simultanée de la touche [Alternate] et d'une des flèches de déplacement du curseur. Les boutons gauches et droits de la souris sont eux remplacés par [Alternate][Insert] et [Alternate][Home]. Le double-clic est obtenu par deux pressions successives et rapprochées de la seconde touche.

☛ **Joystick**

Le ST reconnaît tous les modèles couramment employés sur les consoles de jeux ou les ordinateurs familiaux. On peut employer deux joysticks simultanément. Le second prendra dans ce cas la place et le connecteur de la souris.

☛ **TrackBall**

Il s'agit, à peu de choses près, d'une souris placée sur le dos. Son avantage principal sur celle-ci est de n'occuper sur votre bureau qu'une surface fixe et plus restreinte. S'il veut la remplacer totalement, il doit, comme elle, être muni de deux boutons.

Scanner

Un scanner permet la saisie optique d'une image sur papier (photo, dessin, texte), la rendant ainsi exploitable par l'ordinateur. Il parcourt l'image point par point, ligne par ligne et la stocke en mémoire ou dans un fichier. Il existe des scanners noir et blanc, attribuant à chaque pixel lu une teinte de gris - même si l'image source est en couleur - et des scanners couleurs, qui reconnaissent les différentes couleurs.

Les performances relatives se mesurent en vitesse de lecture, résolution et nombre de couleurs ou de tons de gris reconnus. La résolution se mesure en dpi - dots per inch -, ou points par pouce. Une valeur courante est de 300 dpi.

On trouve pour ATARI plusieurs modèles se différenciant par le mode de lecture des documents :

○ *scanners à plat* : les plus chers et les plus beaux. C'est comme une photocopieuse, mais sans l'imprimante.

○ *scanners à main* : ils ressemblent à une grosse souris et ne peuvent lire qu'une surface réduite en largeur. Ils suffisent pour digitaliser une colonne de texte de journal ou une photo. On les tient à la main, tout comme la souris à laquelle ils ressemblent, et on leur fait parcourir de haut en bas le texte à digitaliser. La largeur couverte varie de 6 à 10 cm. Notons que leur résolution n'atteint que 200 dpi.

○ *scanners pour imprimante*: les plus économiques. Ils sont composés d'une petite cellule de lecture que l'on fixe à la tête de l'imprimante et qui est reliée à l'ordinateur par un câble. Le chariot de

l'imprimante parcourt toute la feuille, comme pour une impression normale, mais sa tête d'impression est inactive et la cellule lit le document ligne par ligne. Ce procédé nécessite une imprimante de bonne qualité, pour obtenir un raccord parfait entre les lignes successives.

☛ **Remarque:**

Il ne faut pas confondre lecture de documents et reconnaissance de caractère (OCR - Optical Character Recognition). Le travail du scanner se borne à "pixeliser" une image papier, qu'elle soit photo ou texte suivi. Pour déchiffrer le texte contenu dans la mosaïque de points, il faut un logiciel spécial de reconnaissance de caractères comme, par exemple, le programme *Augur*.

☛ **Tablette graphique**

Il s'agit d'un support magnétique ou électrostatique sur lequel on écrit et dessine à l'aide d'un stylo électronique relié à la tablette par un câble. C'est la solution idéale pour ceux pour qui dessiner avec la souris donne l'impression de manipuler une brique plutôt qu'un crayon. Elle est irremplaçable pour dessiner à main libre, entrer les coordonnées d'un diagramme point par point ou transférer des dessins préexistant (exemple: carte géographique). Quelques caractéristiques sont à prendre en considération pour le choix d'un tel matériel :

○ **Surface de travail** : les modèles les plus courants sont A3 et A4. Si l'on veut que la tablette représente tout l'écran, il faut pouvoir la basculer horizontalement. Elle doit aussi respecter les proportions : un cercle sur votre feuille ne peut pas devenir une ellipse à l'écran.

○ **Résolution** : nombre de points par mm. Généralement 10, ce qui est plus que suffisant; en A4, cela donne 3000 x 2100 points, bien plus que l'écran n'en peut distinguer.

○ **Distance de travail** : espace autorisé entre le support (tablette) et le crayon électronique. C'est important si vous désirez, par exemple, copier une carte à partir d'un atlas sans être obligé de la détacher au préalable.

○ **Travail avec la souris** : certaines tablettes rendent inactive la souris lorsqu'elles sont branchées. Cela peut être gênant.

○ **Bouton droit** : la plupart des modèles émulent le bouton gauche de la souris - interrupteur sur le corps du stylo ou dans la pointe - , mais pas le bouton droit. Il faut alors un équivalent clavier ou, mieux, utiliser la souris en même temps, si elle est restée active (voir ci-dessus).

○ **Port série** : si la tablette s'y connecte, elle rend impossible son emploi simultané pour la connexion d'un modem ou d'une imprimante série.

○ **Vitesse de lecture** : elle doit être suffisante pour suivre la vitesse de déplacement de votre main sans se faire lâcher.

☛ Digitaliseur vidéo

Un digitaliseur vidéo permet de capturer un signal vidéo d'une source vidéo (caméra, magnétoscope) et de le transformer en une image compatible avec votre ordinateur. La forme la plus courante est une boîte qui se greffe d'un côté sur le port cartouche et de l'autre sur la source vidéo.

Un programme doit le piloter pour effectuer le stockage de l'image sur disque et la conversion de formats. Il faut en effet transformer les couleurs analogiquement dispersées de la source en 16 couleurs distinctes acceptables par l'écran de l'ATARI. Il y a une perte de résolution (320 x 200) et de couleurs (16). Il est aussi possible de travailler en noir et blanc. On manipule alors des teintes de gris qui sont affichées sur le moniteur monochrome comme des effets de trames.

Les images digitalisées sont récupérables par les logiciels de dessin classiques grâce au stockage sur disquette dans un format standard.

Les caractéristiques d'un digitalisateur à prendre en compte sont la vitesse de capture d'une image, la résolution (finesse des détails) et le nombre de teintes de gris ou de couleurs qu'il peut distinguer.

☛ Digitaliseur sonore

Il est au son ce que le digitaliseur vidéo est à l'image vidéo: il le numérise et le rend ainsi exploitable par l'ordinateur. Une fois digitalisé, un son (parole, musique ou bruit) peut être stocké dans un fichier pour être restitué par la suite. C'est le cas, par

exemple, de l'accompagnement musical de certains jeux vidéo.

La restitution peut se faire via le haut parleur du moniteur ou avec un synthétiseur sonore. Ce dernier réalise l'opération inverse du digitaliseur : il recrée le son analogique à partir des données numériques. Le synthétiseur donne un résultat de bien meilleure qualité.

3. Périphériques de sortie

Ils appartiennent à deux classes bien distinctes :

- périphériques d'affichage;
- périphériques d'impression.

AFFICHAGE

☛ Moniteur monochrome

Il en existe deux modèles qui ne diffèrent que par la carrosserie : le SM124 et le SM125. Ils offrent une résolution de 640 x 400 points et possèdent tous deux un haut-parleur intégré.

Les caractères sont dessinés dans une matrice de 16 x 8 points, ce qui autorise 25 lignes de 80 caractères. Avec une barre de menu, une fenêtre, ses deux lignes d'informations et deux ascenseurs, on garde au mieux 20 lignes de 75 caractères.

La fréquence de rafraîchissement de l'image est de 71 Hz, c'est-à-dire qu'elle est entièrement redessinée 71 fois par seconde. C'est un gage de non clignotement de l'écran et par la même de grand confort visuel.

☛ Remarque:

Si une image fixe reste présente trop longtemps à l'écran, elle peut brûler le tube cathodique. Cela signifie qu'elle est "gravée" dans l'écran. Pour éviter ce désagrément, certains logiciels éteignent automatiquement l'écran après un laps de temps. Il suffit, pour le rallumer, de toucher le clavier ou la souris. Dans le domaine public existent plusieurs utilitaires qui remplissent le même rôle. Ce sont des petits programmes que l'on place dans le fichier AUTO et qui se chargent automatiquement lors du boot. Ils s'appellent en général "SLEEPxxx.PRG".

☛ Moniteur couleur

Sa fréquence de rafraîchissement de l'image est synchronisée sur celle du réseau électrique national : 50 Hz pour nous, 60 Hz pour les américains. L'image ainsi obtenue est moins stable que sur un écran monochrome. On peut même remarquer un clignotement certain lorsqu'on utilise le logiciel QUANTUM, qui permet d'afficher simultanément 4096 couleurs à l'écran.

☛ Téléviseur

Nous avons vu qu'il était possible de raccorder un téléviseur couleur directement au ST. Un téléviseur

muni d'une broche PERITEL pourra toujours être raccordé, via la sortie moniteur, si l'on dispose du cordon *ad hoc*. Un téléviseur normal ne peut être utilisé qu'avec un modulateur, interne (STFM) ou externe. Il ne faut cependant pas, dans ce dernier cas, s'attendre à des résultats mirobolants. Un moniteur donnera toujours une meilleure image qu'un poste de télévision.

☛ Écran géant

Si 640 x 400 points est une résolution suffisante pour bien des applications, d'autres s'y sentent à l'étroit. Tout ce qui touche à la PAO profite de l'arrivée d'écrans monochromes présentant une résolution accrue. On trouve ainsi des moniteurs ayant une diagonale de 19 pouces et affichant 1280 x 1024 pixels, soit plus de deux fois la capacité du moniteur monochrome standard. Ils permettent de ne plus restreindre l'image à une partie du document.

Un point d'ombre apparaît cependant : ils ne sont utilisables qu'avec un méga ST. Ils se connectent obligatoirement au bus du 68000 qui, nous l'avons vu, n'est pas accessible sur les 1040 ST et 520 ST.

Un problème soft peut de plus se présenter. De nombreux logiciels, dans leurs premières versions, ne tournent pas correctement avec ces moniteurs. Cela est dû le plus souvent aux programmeurs qui n'ont pas respecté certaines règles de programmation propres au GEM et qui garantissent un fonctionnement correct quelles que soient la dimension et la résolution de l'écran. On peut généralement utiliser le moniteur et l'écran en parallèle, l'un ne contenant qu'une portion de l'image affichée par l'autre.

⇒ **SM124 en A4**

Le moniteur monochrome SM 124 s'il est retourné sur le flanc peut servir de moniteur A4 bon marché. Certains logiciels (*2nd Word*, e.a.) l'exploitent. Cette bizarrerie reste cependant une exception.

⇒ **Transfert photographique direct**

La photographie directe d'un écran est souvent source de problèmes : temps de pose, éclairage, cadrage, etc.. Pour ceux qui en font un usage fréquent, existent sur le marché plusieurs périphériques qui automatisent le processus et fournissent directement soit une photo instantanée soit une pellicule impressionnée.

La "*Palette Polaroid*" se connecte à l'ordinateur. Elle nécessite un programme spécial pour la piloter.

Le "*Freeze Frame*" prend place entre l'écran et le ST. Il travaille de manière transparente et ne nécessite aucun programme spécial. Il suffit d'appuyer sur un bouton pour transférer l'image - couleur ou monochrome - de l'écran vers la pellicule.

IMPRESSION

Approcher l'immortalité. Est-ce possible ? Je ne sais pas. Peut-être qu'en laissant derrière soi quelque chose de tangible, de concret, peut-être qu'alors, oui, on a une petite, une toute petite chance. Ce qui nous amène tout naturellement à parler des imprimantes.

Suivant un mécanisme employé à maintes reprises dans ce livre, nous allons passer en revue quatre types d'impressions :

-
- ↳ imprimante à marguerite
 - ↳ imprimante matricielle
 - ↳ imprimante laser
 - ↳ traceur
-

↳ Imprimante à marguerite

Imaginez une marguerite dont chaque pétale métallique porte à son extrémité un caractère : lettre, chiffre ou autre. Cette techno-fleur est placée sur un chariot qui la positionne horizontalement devant le papier. Un moteur pas à pas la fait tourner pour positionner le bon caractère devant un petit marteau qui le frappe. Le chariot se décale ensuite d'un caractère vers la droite.

□ Avantages

⇒ *La qualité d'impression*

Elle est maximale, comme sur une machine à écrire. On parle de qualité courrier.

⇒ *Changement de police de caractères*

Il suffit de changer la marguerite.

Certains modèles disposent d'un ruban bicolore.

□ Désavantages

⇒ *Jeu de caractères réduit*

Une marguerite comporte généralement entre 90 et 120 caractères. C'est peu, face au 256 connus par l'ATARI ST. Majuscules, minuscules, italiques, indices, chiffres, ponctuations. Il ne reste plus beaucoup de place.

⇒ *Mode texte exclusif*

Il est impossible de faire du graphisme. Certaines marguerites possèdent bien quelques caractères semi-graphiques, mais ils sont très limités et peu pratiques.

⇒ *Changement de police difficile*

Le changement de police au sein d'un même document est pénible, et peu pratique : le logiciel doit marquer un temps d'arrêt pour permettre le changement manuel de la marguerite.

⇒ *Le bruit*

Fini le charme champêtre. La frappe répétée du marteau d'impression devient vite assourdissante. La seule solution est alors d'isoler l'imprimante sous un capot antibruit, aussi cher qu'encombrant.

⇒ *La vitesse*

Ou plutôt la lenteur. Si les marguerites de courses peuvent dépasser les 50 caractères par seconde (cps) - merci pour les tympans -, la moyenne tourne

allègrement autour de 15-20 cps et vous laisse le temps d'aller boire un petit café, pour chaque feuille. Impatients s'abstenir.

☛ **Remarques:**

✓ Comme nos forêts, les marguerites sont en péril. Elles se font supplanter sur leur propre terrain - la qualité courrier - par les imprimantes laser et les nouvelles matricielles (24 aiguilles ou jet d'encre).

✓ L'utilisation d'une imprimante matricielle demande un paramétrage correct du système.

✓ Si vous utilisez un traitement de texte, pas de problème: tous possèdent un driver de marguerite, généralement sous le nom BROTHER.xxx DIA-BLO.xxx.

✓ L'impression d'un fichier à partir du bureau utilise par défaut le port Centronics pour une imprimante matricielle de type EPSON. Il suffit d'employer l'accessoire de configuration d'imprimante (voir le chapitre consacré à GEM : accessoires) et de sélectionner l'option marguerite.

✓ Le boustrophédon : C'est une façon de gagner de la vitesse (cps) sans produire plus de bruit : on imprime une ligne de gauche à droite, et la suivante de droite à gauche. Cela évite un retour de chariot mais demande une mécanique plus précise.

☛ Imprimante matricielle

Chaque caractère est représenté par une matrice - mosaïque - de points. Plus il y a de cases dans sa matrice et plus fine sera sa représentation. L'impression est effectuée par une tête placée, ici encore, sur un chariot. Elle n'imprime qu'une colonne à la fois.

Il existe 3 types d'imprimantes matricielles:

- ☛ à aiguilles;
- ☛ thermique;
- ☛ à jet d'encre.

Imprimante à aiguilles

☐ Principe

Ce sont les plus courantes. La tête d'impression comporte un jeu d'aiguilles superposées qui frappent le papier au travers d'un ruban encreur, colonne par colonne.

☐ Qualité d'impression

Elles disposent habituellement de deux, voire trois, modes d'impression : Draft, NLQ, LQ.

○ En mode *draft* (brouillon), les points se distinguent aisément. A cette moindre qualité correspond une vitesse plus grande. C'est donc ce mode que l'on emploiera pour imprimer un listing ou une première ébauche de lettre.

○ En mode *NLQ* (Near Letter Quality), la tête d'impression effectue, après que le papier a été légèrement remonté, un second passage. Les points se recouvrent partiellement et deviennent plus difficiles à distinguer.

○ En mode *LQ* (Letter Quality) ou qualité courrier, il n'y pas de différence visible avec une imprimante à marguerite. Le caractère semble formé d'un bloc et est d'un noir intense.

○ Le nombre d'aiguilles ne cesse de s'accroître : 7, 8, 9, 12, 24 et on parle de 32. Les plus répandus sont 9 et 24 aiguilles. Regardons ces deux cas de plus près.

✓ *Imprimante à 9 aiguilles*

Neuf est un bon chiffre. Par rapport à sept et huit, il présente un avantage certain: les caractères à jambage descendant - p, j, g, q - sont tracés proprement, sans déformation, et le soulignement peut se faire en même temps que l'impression du caractère, sans passage supplémentaire. Les modes *NLQ* ou *LQ* demandent de faire un, deux ou trois passages supplémentaires, ce qui ralentit la frappe d'autant.

La référence dans cette catégorie est l'*EPSON RX* ou *FX*. Les modèles concurrents sont - presque - tous *EPSON*-compatible. Cela signifie qu'on peut les utiliser comme s'il s'agissait d'une imprimante *EPSON* : même driver, même codes de contrôles, même jeu de caractères.

✓ *Imprimante 24 aiguilles*

Non. Elle ne sont pas superposées en une mais en deux colonnes. Cela offre un double avantage : d'une part, les aiguilles sont un peu plus fines - 12 au lieu de 9 (sur la hauteur d'un caractère) - et, d'autre part, les colonnes sont légèrement décalées en hauteur, ce qui permet d'obtenir en un seul passage le recouvrement partiel des points.

Les imprimantes 24 aiguilles offrent généralement deux qualités d'impression : le draft (un seul passage) et LQ (deux passages). Les aiguilles étant plus fines, on peut employer des polices de caractères plus sophistiquées. La double colonne permet de réduire le temps d'impression d'un facteur deux.

La référence en ce domaine est la NEC P6 ou P7.

☞ **Remarques:**

✓ En première approximation, une 24 aiguilles peut se piloter comme une 9 aiguilles, car leurs jeux de commandes sont très proches. Il ne faut donc pas reconfigurer les paramètres du système - GEM - et on peut récupérer les drivers EPSON. Cela fonctionne même en mode graphique, avec des programmes tels qu'*EasyDraw* ou *Degas*. Le seul désavantage à procéder de la sorte est que vous ne profitez pas pleinement des capacités de votre machine.

✓ En seconde approximation, on s'aperçoit que la copie d'écran - [ALT] + [HELP] - ne fonctionne pas avec les 24 aiguilles. Cela provient de l'OS qui s'évertue à employer lors du *hardcopy* un des rares codes de commande non communs aux deux catégories. Non, ne vous arrachez pas les cheveux. Ce n'est pas grave. On peut y remédier, lors du *boot*, en chargeant un petit programme de conversion. Reste à le trouver, chez

votre vendeur, via un cercle d'utilisateurs ou, en dernier recours, chez l'importateur.

✓ "On n'achète pas un chat dans un sac" ou "le vendeur est bien gentil, mais je préfère vérifier moi-même que ça fonctionne." Un cas parmi d'autres: certaines versions de l'imprimante NEC P2200 - une 24 aiguilles - présentent une incompatibilité partielle avec le ST: la sortie de textes ou de graphiques ainsi que le *hardcopy* via l'émulateur se font correctement, mais l'impression correcte par GDOS (en PAO, par exemple) est impossible. Pour une fois, il n'existe pas de petit utilitaire miracle. Problème. Dans ce cas-ci, la seule solution est d'intercaler une interface spéciale qui corrige ce défaut. Ce conseil est général en informatique : essayez avant d'acheter. N'attendez pas de vous brûler pour vous méfier des flammes.

☛ Imprimante thermique

□ Principe

Les aiguilles sont remplacées par des éléments chauffants. L'impression se fait par chauffage d'un papier spécial, thermosensible. Le résultat n'est pas fameux, et se conserve mal dans le temps.

Une autre solution consiste à faire fondre la couche d'encre d'un ruban spécial, la rendant ainsi collante.

□ Avantages

- ⇒ Le silence est presque total car il n'y a pas de frappe.
- ⇒ Qualité d'impression. En fondant, l'encre forme de petits pâtés qui se recouvrent partiellement, faisant disparaître l'aspect pointilliste de la matrice.
- ⇒ Toucher : l'encre, en séchant, garde une certaine épaisseur.

□ Désavantages

- ⇒ Manque de confidentialité: les zones du ruban où l'encre à fondu laissent clairement apparaître, en blanc sur fond noir, le texte imprimé.
- ⇒ Le ruban ne peut servir qu'une seule fois.

☛ Imprimante à jet d'encre

□ Principe

Les aiguilles sont cette fois remplacées par des buses creuses. L'encre est placée dans une cartouche amovible. De minuscules pistons piézo-électriques la propulsent au travers des buses qui la dirigent vers le papier où elle s'écrase avec splendeur avant de sécher. Certaines emploient de l'encre solide qui est liquéfiée par chauffage.

□ Avantages

- ⇒ Silence : comme ci-dessus, il n'y a plus de frappe.
- ⇒ Qualité d'impression: la finesse des gouttes alliée au recouvrement partiel des pâtés permet d'atteindre la qualité courrier.

⇒ **Toucher** : l'encre solide, en se refroidissant donne du relief au texte. - Inconvénients... encrassement des buses: si on ne fait pas un usage régulier de l'imprimante, l'encre peut sécher dans les buses. L'emploi d'encre solide tend à résoudre ce problème par un préchauffage initial de la tête d'impression.

□ **Désavantages**

⇒ **Papier** : l'encre arrivant à l'état liquide sur le support, il ne peut pas être trop absorbant. Il faut souvent employer un papier spécial, plus cher, avec ce type d'imprimante.

⇒ **Prix de revient** : au papier plus cher s'ajoute aussi le prix des cartouches d'encre. Cet aspect est plus important qu'il n'y paraît.

☛ **Imprimantes laser**

□ **Le mécanisme d'impression**

C'est le même que dans les photocopieuses. L'âme de l'imprimante est un tambour rotatif, recouvert d'une substance photo-électrique qui se charge positivement en présence de lumière. L'image de la page à imprimer est tracée "électrostatiquement" par un faisceau laser de faible puissance. Le "toner" (encre solide sous forme de micro-particules), chargé négativement, est attiré par les charges positives du tambour. Le papier, chargé plus négativement que le tambour, attire à son tour les particules d'encre. La dernière étape est la fixation du toner par chauffage de la feuille. L'opération complète prend moins de temps qu'il ne vous en a fallu pour lire ce paragraphe.

L'impression est silencieuse et rapide : de 8 à 12 pages par minute.

□ L'intelligence

En plus de cette partie optico-mécanico-thermique, une imprimante laser a besoin d'un véritable micro-ordinateur intégré pour la piloter. Il est classiquement constitué (sur les imprimantes haut de gamme) d'un 68000 et d'une RAM de un ou plusieurs Mo. La RAM contient les fontes de caractères, le programme de gestion de l'imprimante et l'image de la page à imprimer qui sera recopiée par le faisceau laser sur le tambour.

□ Caractéristiques

La résolution d'une imprimante laser est de 300 dpi (dots per inch = points par pouce) ou 90000 points par pouce carré. Une page A4, de 21 x 30 cm = ± 98 pouces carrés. Il faut donc ± 8 millions de bits, soit 1 Mo de RAM pour représenter l'image de la page à imprimer.

□ SLM804

C'est le modèle ATARI. A la différence des autres imprimantes laser, elle ne contient que le mécanisme d'impression. Pas de RAM donc, ni de 68000, ce qui explique son petit prix. L'image de la page à imprimer est construite cette fois par l'ordinateur lui-même et conservée dans sa propre RAM. Le transfert, du ST vers l'imprimante, de l'image de la page se fait par le bus rapide DMA.

□ Configuration

Le calcul fait plus haut reste valable ici : il faut disposer d'au moins 1 Mo pour construire la page,

plus l'espace nécessaire au programme qui la construit et aux fontes de caractères. Un Méga ST2 s'impose donc. Si on veut imprimer un document où se mêlent texte et graphisme, il faudra alors utiliser un Méga ST4. C'est la configuration typique pour faire de la PAO : Méga ST4 + SLM804.

□ Copie d'écran

La copie d'écran peut être effectuée par l'imprimante laser. Il suffit pour cela d'installer le programme SDUMP.PRG dans le fichier AUTO. La combinaison de touches [Alternate][Help] enverra alors le contenu de l'écran vers la SLM804. Dans le cas d'une image couleur, l'imprimante utilisera une échelle de gris pour les représenter.

□ Emulateur Diablo 630

Ce programme fait croire à l'ordinateur qu'il est relié à une imprimante à marguerite Diablo 630. Il suffit donc que votre traitement de texte dispose d'un driver d'imprimante Diablo 630 pour qu'il puisse piloter la SLM804.

Notons que l'émulateur ne permet d'utiliser qu'une seule fonte de caractères et dans une seule taille. Cette unique police autorise toutefois l'impression en italique, gras, exposant ou indice.

□ Laser et 1040/520 ST

Il est possible d'employer une imprimante laser à partir d'un 520 ou d'un 1040, mais on ne pourra pas exploiter pleinement ses capacités. La première méthode est de l'utiliser comme une imprimante à marguerite ordinaire, avec l'émulateur Diablo. La seconde méthode est offerte par certains logiciels (le traitement de texte "Le Rédacteur" par exemple). Ils

gèrent eux-mêmes l'impression en envoyant le texte ligne par ligne. C'est plus lent qu'avec un Méga ST et seule l'impression de texte est possible.

□ Autres imprimantes

Il est également possible d'employer d'autres modèles d'imprimante laser. La plupart des logiciels de PAO sont distribués avec un driver d'imprimante *Postscript*. Certains - *Publishing Partner* par exemple - supportent même les *Linotypes*, ce qui permet d'atteindre une résolution de 1200 dpi.

☛ Traceur (plotter)

□ Principe

Un traceur est un appareil qui produit des dessins par traçage sur le papier au moyen de plume, pointe ou stylo à bille. Il est le périphérique de sortie idéal pour le dessin scientifique ou d'architecture. Il est, par contre, particulièrement inadapté dès qu'il s'agit d'imprimer du texte ou une copie d'écran.

Certains traceurs permettent des tracés multicolores et/ou d'épaisseur variable. Différents stylos sont alors situés dans un carrousel ou un magasin, sélectionnés et mis en place par un dispositif commandé par le logiciel de dessin.

□ Caractéristiques

Deux types de construction sont disponibles : feuille fixe ou feuille mobile.

Les traceurs à *feuille fixe* : le support est immobile, fixé au traceur par des liens mécaniques,

électrostatiques ou pneumatiques (suction). Le stylo est posé sur un chariot qui coulisse le long d'un bras qui se déplace lui-même le long de la feuille.

Les traceurs à *feuille mobile* permettent d'employer des supports continus (en rouleau), de longueur quelconque. Comme dans une machine à écrire, la feuille est coincée entre des rouleaux en caoutchouc qui assurent le déplacement vertical du papier. Le long de ces rouleaux glisse le chariot. Il porte le stylo et assure son déplacement horizontal. Un trait oblique est obtenu en faisant simultanément avancer la feuille et le chariot.

□ Les caractéristiques qui distinguent les différents modèles entre eux sont:

- ⇒ Le nombre de couleurs disponibles simultanément.
- ⇒ La taille du support.
- ⇒ La précision du positionnement du stylo ($1/10^e$ de mm.).
- ⇒ La vitesse du tracé (en cm/sec).

Comme tout périphérique, un traceur n'est utilisable que si vous disposez du driver *ad hoc*. Les logiciels de dessins techniques en sont tous dotés.

4. Extensions

- ↳ Blitter
- ↳ Abaq
- ↳ 68881
- ↳ MS-DOS Emulator
- ↳ ST-Accelerator
- ↳ ST-Expander

La plupart des produits cités ci-dessus partagent une caractéristique commune : l'imminence. Ils n'arrêtent pas d'"imminer", ils arrivent, ils sont là. Mais, comme soeur Anne, on ne voit rien venir.

Leurs carrières suivent un schéma désormais classique en informatique: annonce, retard, annonce, retard, ... Lorsque vient enfin la distribution, elle connaît des débuts homéopathiques. Le compte-goutte, quoi. Certains se font attendre comme cela un, deux, trois ans, ou même plus, avant d'arriver. Il faut savoir se faire désirer, mais à ce point, c'est du vice. Ne vous attendez donc pas à tous les trouver chez votre revendeur habituel.

Tous n'arrivent pas sur notre marché, et il vous faudra parfois vous adresser directement au constructeur. Si vous ne pouvez vivre sans, parcourez avidement les petites annonces des revues étrangères. Votre bonheur s'y trouve peut-être déjà. Notre liste n'est pas exhaustive, loin s'en faut. Elle reprend quelques-uns des produits les plus représentatifs et les plus généraux. Vous ne trouverez donc pas mention ici de l'extension qui permet à votre

ATARI ST de piloter une centrale nucléaire ou de l'interface pour lui connecter un pifomètre digital sur 14 bits.

☛ Le Blitter

Le Blitter est un coprocesseur graphique, c'est-à-dire une puce développée dans le seul but d'accélérer l'affichage. Nous avons vu que certaines fonctions du GEM - le scrolling e.a. - n'étaient pas très rapides. Le rôle du Blitter est simple. Chaque fois que le système ou un programme de l'utilisateur fait appel à une de ces fonctions lentes, il la court-circuite et exécute sa tâche à sa place.

L'option "blitter" du menu Option permet de le mettre hors fonction.

☛ ABAQ

Les micro-processeurs

Leur histoire commence chez Intel, avec une puce manipulant des données de 4 bits. Sont ensuite venus les 8 bits, puis les 16 bits et enfin les 32 bits. Parallèlement les processeurs ont connu des versions successives, tournant de plus en plus vite. Actuellement, on en est à 25 Mhz. Ce type de technologie commence cependant à atteindre ses limites. Heureusement, une autre voie existe : les processeurs parallèles.

Les processeurs parallèles

Il y a deux manières d'élever la puissance d'un ordinateur : soit on augmente la puissance de son

processeur central, soit on augmente le nombre de processeurs centraux. Cette seconde solution semble plus séduisante, mais il y a un problème. Si vous désirez créer un système 20 fois plus performant que votre ATARI ST, il n'est pas question de placer vingt MC68000 en parallèle. Cela ne fonctionnera pas, car ils n'ont pas été conçus pour travailler ainsi. Il faut des processeurs spécialement développés pour le parallélisme. Le *Transputer* est de ceux-là.

Le Transputer

En 1983 est apparu le premier modèle : le T212, un 16 bits. Fin 1985, *Inmos* présenta le T414, un 32 bits. Vint enfin le T800. Cadencé à 20 Mhz, il possède une mémoire cache interne de 4 Ko et une unité intégrée de calcul en virgule flottante sur 32 ou 64 bits. Sa puissance théorique est de 12 à 15 *Mips* (Millions d'Instructions par Seconde) et ses performances en calcul de 1,5 *Mflops* (Millions d'opérations en virgule flottante par seconde) en 32 bits et de 1,1 *Mflops* en 64 bits.

De plus, chaque *Transputer* est muni de 4 canaux rapides - *links* - qui permettent de le relier à 4 autres.

Architectures parallèles

Il existe, grâce aux quatre "links", plusieurs manières de relier des *Transputer* entre eux: en anneau, en étoile, en maille ou en hypercube.

L'ABAQ

C'est la carte de Transputer choisie par ATARI. La carte de base comprend un T800, 4 Mo de RAM et 1 Mo de mémoire vidéo. Elle possède également diverses interfaces pour la connexion à un Méga ST et aux divers périphériques (Centronics, DMA, RS232, ...). Ses quatre modes vidéo en font une station graphique de choix. Ils permettent un affichage allant de 1280 x 960 pixels en 16 couleurs parmi 4096 à 512 x 480 pixels en 16 millions de couleurs simultanées. Un blitter couleur accompagne le tout.

La carte possède aussi 3 ports d'extension. Un port peut accueillir soit une carte possédant quatre Transputer munis chacun de 1 Mo, soit une carte mémoire de 20 Mo. La configuration maximale de l'Abaq est donc de 1 Transputer et 64 Mo (12 Mips) ou 13 Transputer et 16 Mo (150 Mips).

Utilisation

Tout ordinateur a besoin d'un système d'exploitation. Le ST a le TOS, et l'Abaq a *Helios*. Il faut aussi des langages de programmation: PASCAL, LISP, BCPL, OCCAM, FORTRAN.

68881

Certaines applications font un usage intensif des opérations arithmétiques. C'est le cas des programmes scientifiques en général, et des logiciels de dessin en particuliers.

Chaque fois qu'un programme demande à l'ordinateur de faire une opération arithmétique complexe - sur les réels -, le 68000 doit exécuter une routine particulière du système. Ce n'est donc pas immédiat.

Un *coprocesseur arithmétique* est une puce spécialement et uniquement conçue dans le but d'exécuter ces calculs délicats. Etant un circuit spécialisé, il le fait, et il le fait bien. Le 68000 lui fournit le ou les opérandes ainsi que la fonction à effectuer. Le coprocesseur exécute le calcul et retourne le résultat au processeur central.

Le ST n'ayant pas été conçu à l'origine pour recevoir un coprocesseur arithmétique, il faut qu'une certaine circuiterie enrobe ce dernier pour les rendre compatibles. Plusieurs firmes ont mis sur le marché ce genre d'extensions. Examinons deux exemples typiques.

1. XCEL, de Xetex. C'est une carte qui se connecte sur le port cartouche. Elle s'adapte donc à tous les modèles de la gamme. Des routines pour le langage C sont fournies en même temps.

2. Weide Electronik vend une carte d'extension d'un type différent qui doit être installée à l'intérieur du ST (tous modèles). Elle aussi est accompagnée de bibliothèques de routines pour plusieurs langages: C, FORTRAN, MODULA II, PASCAL.

☛ Remarques:

✓ Il ne suffit pas d'ajouter un coprocesseur arithmétique à votre système pour le voir passer à la vitesse supérieure. Contrairement au Blitter, qui intercepte directement les commandes qu'il peut exécuter, le coprocesseur arithmétique doit être explicitement contacté par le programme qui veut l'utiliser. C'est pour cela qu'il est fourni avec des bibliothèques - une par langage, ou plus précisément par compilateur - permettant de l'utiliser dans les applications maison. Pour qu'un logiciel du commerce en profite, il faut donc qu'il ait été écrit dans ce but.

✓ L'accroissement de vitesse d'exécution est d'autant plus important que le programme fait un usage intensif des opérations mathématiques. Un traitement de texte ne tirera donc aucun avantage d'un coprocesseur arithmétique.

✓ Le prix de ces produits oscille entre 300 et 400\$.

☛ MS-DOS Emulator

Cette extension est en réalité un véritable ordinateur compatible IBM : on y trouve entre autre un micro-processeur Intel 80xx, de la RAM et des connecteurs pour cartes d'extension. On le relie au ST par une interface, ce qui lui permet d'utiliser tous ses périphériques : souris, écran, lecteur, etc. Le moniteur de l'ATARI ST peut être remplacé par un moniteur IBM via les connecteurs d'extensions de l'émulateur.

☛ ST-Accelerator

C'est un circuit (produit par Strange Systems) qui prend place au coeur de votre ST et remplace le 68000 d'origine par une version plus rapide, le 68000-16, qui tourne à 16 Mhz, c'est-à-dire deux fois la vitesse normale. Il double ainsi la vitesse d'exécution de toutes les applications employant le 68000. Les opérations d'entrée/sortie - chargement de fichier, par exemple - ne sont pas accélérées.

☛ ST-Expander

En première approximation, c'est une grosse boîte (de Strange Systems) qui se glisse sous le moniteur. Elle contient une alimentation, un ventilateur, un soquet pour le blitter et des connecteurs permettant de connecter - tiens, comme c'est bizarre - huit cartes d'extensions sur le bus du 68000.

5. Périphériques divers

- ↳ lunettes stéréoscopiques
- ↳ modem

↳ Lunettes stéréoscopiques

Non, ce n'est pas un de ces gadgets anaglyphiques qui vous fait voir rouge d'un oeil et bleu de l'autre. Il s'agit d'un produit sérieux et peu onéreux constitué d'une paire de lunettes à verres en cristaux liquides raccordées au ST par le port cartouche. Elles fonctionnent dans les trois résolutions et sans perte ni de détail ni de couleurs. Le principe exploite la possibilité d'avoir en mémoire plus d'une image à la fois (voir plus haut). Les logiciels qui supportent ces lunettes génèrent deux images, une pour chaque oeil, en tenant compte de l'écartement interpupillaire et de la distance de vision. A chaque rafraîchissement d'écran, c'est-à-dire 70 fois par seconde en monochrome (50 en mode couleur), un des deux verres devient opaque et le système affiche l'image de l'oeil non occulté. Chaque oeil voit alternativement une image sur deux; la persistance rétinienne fait le reste. Ce produit ne serait rien s'il n'était suivi de programmes travaillant en stéréo. On trouve sur le marché plusieurs logiciels de dessin scientifiques, d'architecture et quelques jeux.

☛ MODEM

MODulateur-DEModulateur. Il permet de faire circuler des données numériques sur le réseau téléphonique. Qui dit données numériques dit bits. Il s'agit donc de faire voyager des suites de bits. Le modem émet une onde porteuse de fréquence et d'amplitude constante sur laquelle se superpose le message (les bits). La vitesse de transmission se donne en bips (bits par seconde).

Les normes européennes sont les suivantes (le premier chiffre indique la vitesse d'émission, le second celle de réception):

⇒ V21	300/300
⇒ V22	1200/1200
⇒ V22bis	2400/2400
⇒ V23	1200/75

A côté des modems connectés directement au réseau téléphonique, on trouve également les modems acoustiques ou coupleurs acoustiques. Ces derniers injectent et reçoivent respectivement le signal sonore sur le réseau par l'intermédiaire d'un haut-parleur et d'un micro sur lequel on pose le cornet de téléphone. Ce type de modem est très sensible à la pollution sonore (ambiance bruyante) et ne convient donc qu'aux transmissions en mode V21 (300 bauds).

LES UTILITAIRES

Un utilitaire est un "collaborateur". Seul, il ne sert à rien. Son seul but est de faciliter l'usage des autres logiciels, d'augmenter le confort de l'utilisateur.

De la kyrielle d'utilitaires de toutes sortes disponibles sur ST, il en est quatre qui sortent du lot:

- Disque virtuel (Ram Disk).
- Spooler.
- Corésideur (Switcher).
- Emulateurs.

☛ Disque Virtuel

Il permet de réserver une partie de la mémoire et de s'en servir comme si c'était une disquette. Ce "faux" lecteur apparaît sur le bureau comme un vrai - une icône - et toutes les manipulations usuelles - sauf le formatage, bien sûr - sont disponibles.

➔ *Avantage* : la rapidité.

Le chargement et la sauvegarde de fichiers sur disque virtuel se fait cent fois plus rapidement qu'avec un lecteur classique. Même le disque dur est dépassé.

➔ *Désavantage* : la volatilité.

Si vous éteignez l'ordinateur, les fichiers stockés sur le disque virtuel s'envolent, car ils sont en réalité installés en mémoire.

☛ Spooler

Si vous lancez l'impression d'un texte ou d'une copie d'écran, votre ordinateur se bloque pendant toute la durée de l'opération. Le rôle du spooler est d'éviter cette attente inutile. Il intercepte votre demande et se charge de l'exécuter en "tâche de fond". C'est une forme restreinte de parallélisme. Pendant qu'il gère l'impression, l'ordinateur n'est pas bloqué et votre application continue son exécution.

Pour mener à bien sa tâche, il opère en deux temps. Lorsque vous envoyez le texte vers l'imprimante, il le capture en bloc et le stocke dans une partie de la mémoire qu'il a réservée à cet usage : le *tampon (buffer)*. Le système croit alors que l'impression est terminée, et redevient disponible pour l'utilisateur. Dans un second temps, il envoie un par un les caractères sauvegardés dans le buffer vers l'imprimante.

☛ Corésideur

Beaucoup d'applications sont découpées en une succession d'étapes répétitives. Une session typique de programmation, par exemple, est une alternance d'éditations et de compilations : on met au point le code source, on le compile, on corrige les erreurs, on recompile, on exécute le code objet, etc. A chaque fois, il faut recharger l'éditeur, ou le compilateur en mémoire, ce qui fait perdre beaucoup de temps. Heureusement, il existe une solution : le "corésideur".

Ce néologisme barbare cache une grande idée : pour ne pas devoir recharger un programme par la suite, il faut simplement ne pas le décharger. La mé-

moire de l'ordinateur est découpée en plusieurs partitions dans lesquelles on installe les différents programmes. Comme GEM est un système d'exploitation monotâche, il n'y a à un instant donné qu'un seul programme actif. Pendant qu'il s'exécute, les autres sommeillent. Une combinaison de touches au clavier permet de changer de partition. Le programme courant s'endort, et un autre devient actif. Quand vous reviendrez dans cette partition, vous retrouverez votre programme dans l'état où vous l'avez quitté.

☛ Emulateurs

Un émulateur est un produit qui fait fonctionner l'ATARI ST comme une machine totalement différente. Si un programme dont vous avez besoin existe sur la machine X mais pas sur le ST, il suffira d'un émulateur "X/ST" pour pouvoir l'employer sur votre ordinateur.

Emulateur Macintosh

Le Macintosh, d'Apple présente de nombreuses similitudes avec l'ATARI ST : même processeur central, même taille de disquette, même affichage graphique monochrome et environnement WIMP. On peut pousser la ressemblance plus loin. Pour cela, il suffit de brancher sur le port cartouche un émulateur Macintosh, Aladin ou Magic Sac.

→ *C'est mieux chez nous !*

Tout d'abord, l'écran. Nous, c'est 640 x 400. Lui, c'est 512 x 342. En arrondissant, cela fait 46% d'image en plus.

Ensuite, la rapidité. En moyenne, un programme Macintosh gagne 20% en vitesse en tournant sur un émulateur ST. C'est le résultat de l'architecture interne du ST.

→ **Limite**

Tout cela semble merveilleux, mais il y a un problème: le lecteur de disquette. Une disquette 3'1/2 écrite sur un Macintosh est strictement illisible par un ST. Il faut donc d'abord transférer, via un câble, les données d'une machine vers l'autre. Elles sont ensuite stockées sur une disquette au format ST. Ce transfert obligatoire d'une machine vers l'autre est impossible à réaliser avec les logiciels protégés. Ce problème sera résolu quand des lecteurs Macintosh seront disponibles pour le ST.

Emulateur IBM-PC

✦ **PC-Ditto**

Il s'agit d'un programme qui émule un IBM-PC. Une fois lancé, il fait réagir l'ATARI ST comme un IBM-PC que l'on vient d'allumer. Il faut donc commencer la session de travail en chargeant le système d'exploitation MS-DOS.

➤ **Format des disquettes.**

Nous avons vu dans le chapitre OS que l'ATARI ST et les compatibles IBM-PC partagent un même format de disquette. Les informations sont stockées de la même manière par le TOS et par MS-DOS.

Une certaine discipline est cependant nécessaire pour échanger sans problème des fichiers entre les deux

machines car le ST est moins pointilleux que les PC. Il accepte que l'on prenne certaines libertés. Nous pouvons ainsi formater une disquette en augmentant le nombre de pistes et en mettant plus de secteurs sur chaque piste. La capacité d'une disquette peut ainsi passer des 720 Ko réglementaires (en double face) à plus de 820 Ko sans que notre système ne rechigne. Mais elle sera alors devenue illisible pour les PC. Que faire donc ?

Marche à suivre.

1° Formater la disquette sur un PC en simple ou double face.

Note : il existe de nombreux utilitaires qui permettent de formater correctement (aux normes MS-DOS) une disquette sur ST.

2° Installer le *boot sector* MS-DOS.

La disquette est ainsi utilisable par les deux machines. Vous pouvez l'employer pour échanger :

- ✓ Des fichiers ASCII pur.
- ✓ Des fichiers "software-compatible": le traitement de texte *WordPerfect*, par exemple, existe sur les deux machines et utilise un même format. Un texte mis au point avec sa version PC pourra être récupéré et utilisé sur sa version ST.
- ✓ Des programmes pour PC que l'on exécutera via l'émulateur PC-Ditto.

► *La lenteur*

Comme les processeurs centraux et donc les langages machines des deux ordinateurs sont différents, chaque instruction pour l'Intel 8088 doit être traitée individuellement et son fonctionnement doit être simulé par le 68000 du ST. Cela prend plusieurs

temps d'horloge, ce qui explique qu'un programme sur PC-Ditto tourne en moyenne trois fois moins vite que sur un vrai PC. Ce problème est résolu par l'émulateur hard MS-DOS (voir le chapitre consacré aux *PÉRIPHÉRIQUES* : extensions).

Autres émulateurs

Emulateurs de processeurs

Citons encore les émulateurs ATARI XL, CP/M, ATARI 800, COMMODORE 64 et APPLE II. (*Note* : les deux derniers n'émulent que les programmes Basic.)

Emulateur d'écran

C'est un programme qui se charge lors du boot et permet d'utiliser avec votre moniteur couleur (monochrome) des programmes travaillant en monochrome (couleur). Il faut ajouter que les programmes ainsi émulés sont fortement ralentis.

LES VIRUS

Dans cette époque viralemment trouble, la contagion a même atteint nos machines. Pas de pitié pour les puces. Si le ST n'a pas compté ce type d'infection parmi ses maladies de jeunesse (il en a fait bien d'autres), il semblait inévitable que son succès croissant attire les contamineurs de tous bords, plus ou moins vicieux dans leurs créations perverses.

Si vous copiez des logiciels obtenus par la bande, qu'ils soient piratés ou non, vous faites partie de la population à risques. La méfiance en ce domaine est la mère de vos disquettes et à plus juste titre de votre disque dur si vous en possédez un.

☛ Les trois classes de virus.

Il existe trois types de virus:

- ⇒ Les faux programmes.
- ⇒ Les boot-virus.
- ⇒ Les chevaux de Troie.

Analysons leur forme, les risques encourus et les traitements ou précautions à prendre.

⇒ Les faux programmes

Sous le couvert d'un nom anodin ou de prétentions courantes (Exemples: "demo.prg" ou "format.prg"), se cache parfois une farce plus ou moins bonne. Ils

sont faciles à réaliser et peuvent être plus dangereux qu'il n'y paraît.

Il en existe 2 sous-classes :

Les gentils : ils se signalent directement (à l'exécution), d'une manière évidente, et ne détruisent rien.

Les méchants : ils installent sournoisement un morceau de code pirate qui se tapit dans un coin de la mémoire et y reste jusqu'à l'extinction de la machine. Il peut alors se livrer en secret aux pires exactions. C'est ce qu'il faut détecter le plus vite possible.

☉ **Remarque :**

Ces programmes ne sont pas auto-reproducteurs. Il suffit dès lors de les repérer au plus tôt et de les éliminer. Ils - les méchants - peuvent cependant installer une routine qui créera un virus d'un des deux autres types.

☞ **Les boot-virus :**

Le boot-sector :

Lorsque l'on démarre le système, en allumant la machine ou après un *reset*, le TOS va consulter le secteur 0 piste 1 de la face 0 du lecteur "A" (ou éventuellement du disque dur). Ce secteur s'appelle le "*boot-sector*". Il permet d'automatiser certains processus. Si vous employez un système d'exploitation sur disquette, c'est le *boot-sector* qui ordonnera automatiquement son chargement en mémoire à l'allumage de la machine. De même, certains programmes du commerce l'emploient comme protection anticopie ou par commodité. Il suffit d'allumer la machine avec la disquette dans le lecteur

"A", et tout s'installe automatiquement. C'est le cas de la plupart des jeux.

Le boot-virus :

Ce programme qui s'exécute automatiquement et d'une manière invisible pour l'utilisateur peut très bien être un virus; il suffit que la tâche exécutée soit maligne et le tour est joué. C'est dans cette classe que l'on rencontre la majorité des virus. Le scénario est toujours le même:

➤1. *L'installation*

Lors du *boot*, le programme pirate s'installe en mémoire à une place qui lui permettra d'agir à sa guise et en toute indépendance: sur une des routines d'interruption du système. L'interruption *vbl*, par exemple, déclenche un ensemble de routines soixante-dix fois par seconde. Il suffit au code parasite de s'installer dans cette liste de mini-programmes pour être réveillé périodiquement et agir.

➤2. *La reproduction*

Une fois installé, le virus cherche à se reproduire, c'est-à-dire s'installer dans le *boot-sector* des disquettes propres. Il surveille le lecteur de disquettes, à l'affût du changement de disque. Une fois le changement effectué, et repéré, il lui suffit de copier son code originel dans le *boot-sector* vierge. Cette opération prend environ deux secondes (cas du lecteur) et se signale par une activité du lecteur de disquettes. Elle n'est évidemment possible que si la protection en écriture est levée.

➤3. *L'action proprement dite*

Elle peut être plus ou moins agressive (formatage, message, destruction de données,...), et son apparition peut être retardée. Certains, par exemple, attendent d'avoir infesté un certain nombre de nouvelles

disquettes avant de passer à l'attaque. C'est d'ailleurs un gage d'efficacité.

Traitement :

Heureusement pour le monde libre, certains programmeurs ont réagi, face à la menace, en créant des programmes tueurs de virus. Le principe en est simple :

1. *Dépister le virus* : si les deux premiers octets du secteur 0 piste 1 contiennent "6038" (c'est-à-dire une instruction de saut en assembleur), il y a un programme dans le *boot-sector*.
2. *Tuer le virus* en redonnant au secteur malade sa virginité originelle.



Attention danger : comme dit précédemment, de nombreux programmes utilisent correctement le *boot-sector* et son intégrité est donc vitale à leur bon fonctionnement. Certains traitements, étant pires que la maladie, pourraient ainsi endommager définitivement des disquettes saines et rendre inutilisable leur contenu. Prudence donc si vous décidez de soigner vos disquettes.

☞ **Remarque:**

Ce type de virus ne peut s'attraper que si la disquette contaminée sert au boot.

⇒ Le cheval de Troie

Structure d'un fichier programme :

Lorsque un programme est chargé en mémoire, le système doit disposer de certaines informations pour

mener à bien cette mission : taille du code du programme, première instruction exécutable, etc. Ces données sont localisées dans le *header*, petit bloc d'informations système, placé en tête du programme. Un fichier programme sur une disquette est donc composé de 2 parties: une courte, le *header*, suivie immédiatement par le programme proprement dit.

Le virus :

Il infeste les programmes sains en se collant à eux. La contamination consiste à ajouter au code du programme, sur disquette, une routine parasite. Le *header* est mis à jour en conséquence : nouvelle taille du programme total (plus grande) et le pointeur vers la première instruction à exécuter est dirigé vers le début du virus. De la sorte, chaque fois que l'on charge un programme contaminé, le virus est exécuté en premier, avant qu'il ne passe la main au programme principal. Il s'arrange par la suite pour se propager, c'est-à-dire se greffer sur de nouveaux programmes sains.

Traitement :

Il n'y a pas de solution simple à ce problème. Même le dépistage en est très difficile. Il faut surveiller la taille de vos programmes pour repérer ceux qui grossissent mystérieusement. Par chance, ce type de virus n'a pas (encore) atteint le ST. Mais on ne sait jamais. Un utilisateur averti en vaut deux.

Conseils généraux :

- ⇒ Protégez vos disquettes en écriture. Cela empêche le virus de se propager.
 - ⇒ Toujours booter à partir d'une disquette sûre.
 - ⇒ Débranchez le disque dur quand vous essayez un programme suspect.
 - ⇒ Vérifiez périodiquement le *boot-sector* des disquettes.
 - ⇒ Il est possible de rendre un virus résistant au *reset*. Si vous détectez une anomalie, éteignez votre machine et attendez une quinzaine de secondes avant de la rallumer.
 - ⇒ Ne sous-estimez pas le problème. Comme pour son homonyme médical, le virus informatique s'attaque surtout à ceux qui ne se méfient pas.
-

LE HARDWARE

Le "hardware", par opposition au software désigne tout ce qui est dur, concret, ce que l'on peut toucher. Il s'agit entre autres des circuits de l'ordinateur et de son architecture interne. Bien qu'étant aussi durs et concrets, les périphériques seront traités dans un chapitre séparé. Ici seront donc abordés les constituants internes de l'ATARI ST et ses différentes interfaces vers les périphériques.

☛ LE MC68000

L'ATARI ST est équipé d'un MC 68000 tournant à 8 Mhz. C'est un 16/32 bits : il calcule en interne sur 32 bits, mais les échanges de données avec la mémoire se font par paquets de 16 bits. Son bus de 24 bits lui permet d'adresser une mémoire de 16 Méga-octets (Mo). Il offre à l'utilisateur 16 registres de 32 bits.

☛ La ROM

C'est la mémoire morte. Elle contient dans ses 192 Ko le système d'exploitation et le GEM. Elle est uniquement lisible (Read Only). Son contenu est fixé en usine et non modifiable par l'utilisateur.

☛ La RAM

Mémoire Vive. Elle s'étend suivant les modèles de 252 Ko à 4 Mo. Bien que le MC68000 puisse

adresser 16 Mo de mémoire, une particularité - autrement dit une erreur de conception - du ST la limite à un maximum de 4 Mo.

☛ **Le clavier**

Que peut-on bien dire à propos du clavier ? Avec un autre ordinateur, rien, si ce n'est énumérer les touches et leur signification et ajouter que le processeur central va sans cesse regarder si on n'en presse pas une. Avec l'ATARI ST, grâce au 6301, c'est différent.

Le 6301 est un processeur de la famille 6800 de Motorola. Il s'agit d'un "single-chip computer", c'est à dire un véritable petit ordinateur dans un seul circuit intégré. Il possède ses propres ROM et RAM et est uniquement dédié à la scrutation du clavier. Dès que vous pressez une touche, il la décode et avertit le processeur central. Il lui évite donc la scrutation perpétuelle et est ainsi un gage de rapidité accrue. En plus de la surveillance du clavier, il s'occupe aussi de la souris et du joystick. Lorsque vous déplacez la souris sur votre bureau, il calcule sa nouvelle position et envoie directement au processeur central ses coordonnées en X et Y.

☛ **La souris**

Son fonctionnement est assez simple. La bille caoutchoutée qu'elle contient est en contact avec deux molettes, orientées à 90° l'une par rapport à l'autre qui suivent les mouvements en X et en Y. Ces deux molettes entraînant deux petits disques perforés qui

font office de barrières lumineuses. Lorsque vous déplacez la souris, les deux molettes interrompent plus ou moins les signaux lumineux générés par quatre diodes. Ces variations sont transmises au 6301 qui en déduit le déplacement absolu de la souris et donc sa position nouvelle. En plus des déplacements, elle peut aussi transmettre les pressions effectuées sur ses deux boutons dorsaux.

☛ L'interface Centronics

C'est l'interface parallèle où l'on peut connecter une imprimante ou un plotter - table traçante - Centronics standard. Son paramétrage se fait avec l'accessoire "Install Printer".

☛ L'interface RS-232

C'est l'interface série. Elle permet de connecter un modem ou de se relier à un autre ordinateur. Bien qu'on puisse aussi y connecter une imprimante - ou un plotter - série, il faut signaler qu'un bug dans l'OS - ancienne version - le rend impropre à cet usage.

Les vitesses de transfert des informations par la prise RS-232 disponibles sont 300, 1200, 4800 et 9600 bauds ou bits par seconde. Au-dessus de 9600 bauds, les parasites perturbent le transfert. L'OS fournit les fonctions de gestion nécessaires à l'utilisation de cette interface et deux protocoles de communication - XON/XOFF et RTS/CTS.

Le paramétrage de l'interface peut se faire au moyen de l'accessoire "Set RS-232 Config". Il s'utilise

de concert avec l'accessoire "Install Printer" qui permet, lui, de spécifier au système que l'on emploie une imprimante série.

☛ Les prises MIDI

La communication entre instruments MIDI est basée sur la transmission en série par boucle de courant. Cette technique est peu sensible au bruit et permet d'atteindre une vitesse de 31250 bauds, ce qui est plus de trois fois la vitesse maximum du RS-232.

La prise MIDI-IN reçoit les données. Un optocoupleur isole électriquement l'ordinateur de la ligne MIDI, ce qui le rend insensible à une éventuelle surtension dangereuse - pour ses circuits - qui pourrait apparaître sur cette ligne.

La prise MIDI-OUT injecte les données produites par l'ordinateur dans la ligne MIDI.

Les instruments MIDI possèdent une troisième broche, absente sur le ST : MIDI-TRU. On y retrouve les informations qui entrent dans l'ordinateur. Cette lacune n'est pas bien grave car notre broche MIDI-OUT cumule ces deux fonctions, OUT et THRU. Il suffit donc d'un bricolage minime - séparer les fonctions cumulées - pour installer la troisième broche sur le ST.

☛ Le port Cartouche

Il sert exclusivement à l'insertion de cartouche de ROM externe de 128 Ko maximum. En théorie. Certaines astuces permettent de l'employer dans l'autre sens - l'écriture - et de dépasser cette limite.

C'est le cas entre autres des *Solid State Disk* qui permettent d'y stocker jusqu'à 1 Mo d'informations dynamiques. (voir le chapitre *PÉRIPHÉRIQUES : Stockage*).

Bien d'autres périphériques encore le détournent de son utilisation initiale pour se connecter à l'ordinateur. Citons les digitaliseurs sonores et vidéo, les horloges permanentes externes, les lunettes stéréoscopiques, les scanners d'imprimantes, etc.

L'avantage de cette connexion "pirate" est sa rapidité de transfert d'information. Elle est proche de celle du transfert de mémoire à mémoire, autrement dit quasi instantanée.

☛ Le contrôleur de disquette

Le circuit employé pour piloter le lecteur de disquettes est un WD1772 de Western Digital. Il est compatible avec celui employé dans les IBM-PC et autres clones. Bien qu'il laisse à l'utilisateur la possibilité de formater autrement une disquette - e.g. longueur d'un secteur, nombre de pistes et secteurs -, c'est le format standard qui est optimum en ce qui concerne le transfert de données.

Il travaille aussi bien avec des disquettes 3"1/2 que 5"1/4 et ce en simple ou double densité. Il inscrit dans chaque secteur une information supplémentaire - le *CRC : code de redondance cyclique* - qui lui permet, lors de la lecture ultérieure du secteur, de détecter une éventuelle erreur qui en aurait modifié le contenu.

☛ LE YM-2149.

Ce circuit comporte trois générateurs de sons et un générateur de bruit indépendants et programmables. Le son produit s'étend sur huit octaves et quinze niveaux de volume. Il est envoyé directement dans le haut-parleur du moniteur ou du téléviseur.

☛ L'interface DMA

C'est une interface parallèle ultra-rapide. On peut y connecter de un à huit périphériques DMA. Cela concerne le disque dur, l'imprimante laser, le CD-ROM, etc..

La transmission se fait à une vitesse maximale de 1 Mo par seconde. Cette rapidité extrême est due à un coprocesseur spécialisé dans le transfert de données entre la mémoire et un périphérique externe.

Examinons sommairement son fonctionnement lors de la lecture de données sur le disque dur dans deux cas : sans le DMA - la majorité des ordinateurs actuels - et avec DMA (cas de l'ATARI ST). Dans chaque cas, le but est le même : copier les informations en provenance du disque dur dans une partie de la mémoire.

➡ **Sans DMA.** Chaque donnée - octet - en provenance du disque dur est copiée par le MC68000 dans un de ses registres. Un autre de ses registres contient l'adresse de destination de l'octet lu. Il y est recopié. L'adresse est incrémentée. Et le processus recommence. Le trajet des données est donc :

Disque dur -> registre -> mémoire

⇒ Avec DMA. Le processeur central indique au circuit chargé du DMA l'adresse où transférer les données. Celui-ci y recopie directement l'information en provenance du disque dur sans stockage intermédiaire. Le trajet des données est donc cette fois:

Disque dur -> mémoire

Cela explique la dénomination de cette technique : Direct Memory Access (accès direct à la mémoire)

En raison de la grande vitesse du transfert, le câble qui relie un périphérique DMA à l'ordinateur ne peut pas être trop long, pour éviter les parasites.

ANNEXES

<p>☛ Notions de base</p>

Un ordinateur a pour tâche le traitement de l'information. A cette fin, il dispose de :

- ✓ un *micro-processeur* central : son intelligence.
- ✓ une zone de *mémoire* où stocker les données des problèmes à résoudre.

Pour mener à bien sa tâche, il a de plus besoin d'une description détaillée de celle-ci : c'est le *programme*. Les données à transformer pour obtenir les résultats sont contenues dans un *fichier*. Celui-ci est un ensemble d'informations (texte, images, données chiffrées, etc..) regroupées et sauvegardées sur un support magnétique : la *disquette*. Pour lire et écrire les informations sur cette disquette, l'ordinateur se sert d'un *lecteur de disquettes*.

Une fois obtenus par traitement des données, les résultats peuvent être affichés sur un écran ou imprimés. Tous les appareils que l'on peut connecter à l'ordinateur sont repris dans le terme générique de *périphériques*.

Pour un chiffre - dans le système décimal -, dix valeurs sont possibles, de 0 à 9. L'ordinateur n'emploie pas le système décimal, mais bien le *système binaire*. Dans ce système, seules deux valeurs sont possibles pour un chiffre : 0 ou 1. Elles correspondent au passage ou non d'un courant électrique dans les circuits. Un chiffre binaire se dit en anglais "binary digit", que l'on a raccourci en "bit".

Le *bit* est la plus petite unité d'information traitable par les circuits de l'ordinateur. Huit bits successifs forment un *octet*. L'octet est l'unité de mesure de la taille de la mémoire. Deux octets - 16 bits - forment un mot et deux mots - 32 bits - forment un long-mot.

☛ Liaison série/parallèle

Il y a deux manières d'envoyer un caractère - 8 bits - vers l'imprimante : en série ou en parallèle. On peut comparer la première à une route à voie unique et la seconde à une autoroute à huit bandes - ou plus.

Dans la *liaison parallèle*, les huit bits sont envoyés d'un coup vers l'imprimante. Ils voyagent chacun en parallèle sur un câble individuel. Pour une telle transmission, il faut donc au moins huit fils - un par bit.

Dans la *liaison en série*, un seul fil est consacré au transport de l'information. Pour transmettre un caractère, il faut donc le décomposer en ses huit bits et les envoyer à la queue leu leu sur l'unique câble.

Ces deux types de transmission de données s'appliquent aussi à d'autres périphériques : le modem et le lecteur de disquettes (série) ou disque dur et l'imprimante laser (parallèle + DMA).

☛ Notations

• Mémoire

Ko = kilo-octets = 1024 octets

Mo = méga-octets = 1024 Ko.

• Clavier

[SPACE] = appui sur la barre d'espace du clavier

[RETURN] = appui sur la touche Return

[CONTROL A] = appui simultané sur la touche CONTROL et la lettre A.

• Disquettes

SF = simple face

(**SS** = single side)

DF = double face

(**DS** = double side)

SD = simple densité

(**SD** = single density)

DD = double densité

(**DD** = double density)

• Imprimante

Dpi = dots per inch : points par pouce

1 pouce = 2.54 cm.

Cps = caractères par seconde

1 caractère = 8 bits.

• Modem

Bips = bits par seconde

1 baud = 1 bit

☛ Acronymes

OS	Operating System (système d'exploitation)
GEM	Graphics Environment Manager
TOS	Tramiel Operating system
RAM	Random Access Memory
ROM	Read Only Memory
DMA	Direct Access Memory
PAO	Publication Assistée par ordinateur (en anglais, Desktop Publishing)

☛ Synonymes

- Application ou logiciel ou programme.
- Ecran ou moniteur.
- Lecteur ou lecteur de disquettes.
- Puce ou circuit.
- Processeur central ou microprocesseur ou MC68000.
- Cliquer ou sélectionner.
- [RETURN] ou [ENTER].

TABLE DES MATIÈRES

Sommaire	5
Introduction	7
Généralités	11
La gamme ST	13
Un peu d'histoire	13
Une idée	14
Un peu de politique	14
Performances	15
Complétude	15
Confort	21
Uniformité	22
1. Les points communs	23
2. Les différents modèles	24
130 ST	25
260 ST	25
520 ST	25
1040 ST	27
MEGA ST2/ ST4	28
3. Différences pour l'utilisateur	30
3.1. Les différences en taille mémoire	30
3.2. Les différences de lecteurs	33
3.3. Les différences du nombre de lecteurs	35
3.4. Les différences d'écran	36
1. Couleur ou monochrome	37
2. Exclusivement monochrome	37
3. Exclusivement couleur	38
3.5 Les différences nationales de l'OS	39
1. Le clavier	40
2. Les messages du GEM	41
3.6 Les versions successives de l'OS	42

3.7 L'accès au bus	43
4. L'homogénéité et ses avantages	43
Ouverture MS-DOS	44
Le système d'exploitation	45
Késako	47
Ses rôles	48
Gestion du disque	49
Un fichier	50
Répartition des morceaux	51
La fragmentation	52
Le pire des cas de fragmentation	53
Remèdes anti-fragmentation	53
Format étendu	55
Autres OS	55
Conséquences	56
Pourquoi un autre OS ?	56
OS multi-tâche	56
OS multi-utilisateur	56
OS multi-tâche et/ou multi-utilisateur	58
Des noms, des noms	59
GEM	61
Non	63
W.I.M.P. 63	
GEM	63
1. Le GEM pour l'utilisateur	65
Premiers contacts	65
Première fenêtre	68
Le catalogue	73
Manipulations de fichier	80
Le menu fichier	85
Le menu Options	89
Le menu Bureau	95
Bombes, fenêtres et icônes	102

2. Le GEM pour le programmeur	106
V.D.I.	106
A.E.S.	109
3. L'ATARI sans GEM	111
Les langages de programmation	115
1. Notions générales	117
Programme	117
Langages et générations	117
Langages de 1 ^{re} génération	118
Langages de 2 ^e génération	118
Langages de 3 ^e génération	119
Langages de 4 ^e génération	120
Interpréteur / compilateur	120
Editeur	123
Debugger	126
Ressources	127
Mélange de langages	127
Pourquoi différents langages	128
2. Les différents langages	130
L'Assembleur	131
C	133
BCPL	135
FORTRAN	136
BASIC	137
PASCAL	139
MODULA-2	141
LISP	143
APL	146
FORTH	149
OCCAM I	151
Le graphisme	155
1. Le matériel	157
L'équipement	157

Les couleurs	158
Les trois résolutions	159
L'écran graphique	160
Le principe du bitmap et de la palette	161
Le principe bitmap	161
La palette	163
Le bitmap en deux plans	164
2. Logiciels de graphisme	166
D.A.O orienté pixel	166
DEGASELITE	166
SPECTRUM 512	167
QUANTUM	168
D.A.O. orienté objet	169
EASY DRAW	169
CAD3D	171
Animation d'images	173
CYBERCONTROL	174
CYBER PAINT	174
CAO.	174
Le relief	175
Les traitements de textes	177
1. Introduction	179
2. Termes usuels	179
3. Remarques communes	183
4. Les quatre types de programmes	185
Editeur	185
Traitement de textes	187
Traitement de documents	191
P.A.O.	192
La musique et la norme MIDI	195
Circuit sonore	197
Son digitalisé	197
M.I.D.I.	197

Les périphériques	201
1. Périphériques de stockage	203
Lecteur de disquettes	203
Disque dur	206
FD-10 208	
CD-ROM	210
Solid State Disk	213
2. Périphériques d'entrée	214
Clavier	214
Souris	215
Joystick	215
Trackball	215
Scanner	215
Tablette graphique	217
Digitaliseur vidéo	219
Digitaliseur sonore	219
3. Périphériques de sortie	221
Affichage	221
Moniteur monochrome	221
Moniteur couleur	222
Téléviseur	222
Ecran géant	223
Transfert photographique direct	224
Impression	225
Imprimante à marguerite	225
Imprimante matricielle	228
Imprimante à aiguilles	229
Imprimante thermique	231
Imprimante à jet d'encre	232
Imprimante laser	233
Traceur	236
4. Extensions	238
Blitter	239
ABAQ	239

68881	242
MS-DOS Emulator	244
ST-Accelerator	244
ST-Expander	244
5. Périphériques divers	245
Lunettes stéréoscopiques	245
MODEM	246
Les utilitaires	247
Disque virtuel	249
Spooler	250
Corésideur	250
Emulateurs	251
Emulateur Macintosh	251
Emulateur IBM-PC	252
PC-DITO	252
Autres émulateurs	254
Les virus	255
Les trois classes de virus	257
Les faux programmes	257
Les boot-virus	258
Le cheval de Troie	260
Conseils généraux	262
Le hardware	263
Le MC68000	265
La ROM	265
La RAM	265
Le clavier	266
La souris	266
L'interface Centronics	267
L'interface RS-232	267
Les prises MIDI	268
Le port Cartouche	268

Le contrôleur de disquette	269
Le YM-2149	270
L'interface DMA	270
Annexes	273
Notions de base	275
Liaison série/parallèle	276
Notations	277
Acronymes	278
Synonymes	278

La chaîne Atari-PC-Mac: du Rédacteur à Word3.

Ce volume a été entièrement saisi sur ATARI ST, à l'aide du traitement de texte *Le Rédacteur*, puis transféré au format ASCII sur une disquette formatée par un IBM-PC.

Le transfert PC-MACINTOSH (au format *Mac Write*) a été possible grâce à *MacLink Plus*. La mise en page définitive a été faite sur MACINTOSH II à l'aide de *Word 3.01* (lequel importe directement les fichiers *Mac Write*).

Les polices utilisées sont Times, Monterey, Mobile, Cairo, Avant-Garde, Lubalin, Courier et ThinTimes.

L'impression a été faite sur LASERWRITER II NTX, à l'exception des illustrations qui ont été imprimées sur la laser d'ATARI.

La réalisation est due aux *Ateliers Virga Processing*.

IMPRIMÉ EN FRANCE PAR BRODARD ET TAUPIN
1914A-5 - Usine de La Flèche (Sarthe), le 28-11-1988.

pour le compte des
Nouvelles Editions Marabout
D.L. décembre 1988/0099/253
ISBN 2-501-01094-9

L'indispensable pour Atari

Matériel et logiciels

Atari est un micro-ordinateur très largement répandu en France dont il existe plusieurs modèles qui se différencient uniquement par la taille de la mémoire centrale.

Le micro-ordinateur Atari est, à l'heure actuelle, celui qui présente le meilleur rapport qualité/prix. La chaîne de micro-édition construite autour d'un Atari est moitié moins chère qu'une chaîne équivalente configurée autour d'une autre marque.

En utilisant certains programmes, on peut rendre cet ordinateur IBM compatible et aussi Macintosh compatible.

Ce livre, véritable petite bible de l'Atari tant du point de vue du matériel que des logiciels, s'adresse bien sûr à tous ceux qui possèdent cet ordinateur, mais aussi à ceux, professionnels ou non, qui s'interrogent sur toutes ses possibilités.

L'auteur:

A. Ravet est licencié en informatique et possède depuis plusieurs années un Atari qu'il a soumis à tous les tests ludiques et scientifiques.

40 3861 8



9 782501 010948



marabout
SERVICE

