

CeFIS - FUNDP NAMUR

Année 2001-2002

FORMATION J

Tout (en partant de rien) pour utiliser les ordinateurs

(Séances 1 et 2)

***Pour une utilisation efficace
des technologies de
l'information***

Notions de base

Etienne VANDEPUT

OBJECTIFS

Cette formation s'adresse (de préférence) aux enseignants qui ont envie d'intégrer l'outil informatique dans leur pratique professionnelle (rédaction de notes de cours, présentations interactives, gestion d'informations diverses: listes d'élèves, d'exercices, de questions..., utilisation de l'ordinateur comme super-tableau, exploitation des ressources d'Internet, du courrier électronique...) et qui ne le connaissent pas ou si peu.

Pour parler clairement, la première partie de cette formation n'a certainement pas comme objectif de faire acquérir aux participants toutes les connaissances et toutes les compétences nécessaires à cette intégration. **Son but est plutôt de donner une idée précise de ce que représente le travail avec un ordinateur et des logiciels, comme démarche intellectuelle, à travers un mélange d'activités pratiques et de réflexions communes plus théoriques.** Elle met en garde contre des tas d'idées reçues telles: "en cinq minutes vous avez tout compris", "les enfants utilisent très facilement un ordinateur", "le travail avec un ordinateur ne demande pas de réflexion", etc. Au terme de ces deux premières séances, les participants devraient avoir pris conscience de la complexité du travail avec un ordinateur mais aussi, s'être forgé une représentation correcte de son fonctionnement de nature à améliorer l'efficacité de leur travail futur avec lui.

Cette première réflexion aura pour objet de faire comprendre le dessous des cartes, dans une raisonnable mesure, pour aider à fonctionner en utilisateur averti, organisé et efficace. Elle se fera par le biais et les manipulations de base de plusieurs "outils": un environnement graphique de travail (celui de *Windows* version 95 ou supérieure), un logiciel de dessin par points (*Paint*), un logiciel de dessin vectoriel (*Corel Presentations*) et quelques petits utilitaires disponibles à partir de l'environnement *Windows*.

L'esprit de cette formation est de respecter le rythme des participants, mais aussi de répondre à toutes les questions qui surgiront ou qui sont déjà présentes dans leur tête. Il ne s'agit pas de leur faire ingurgiter des connaissances volatiles (tout change tellement vite à ce niveau) mais de leur faire entrevoir des modes de travail et de leur fournir des images mentales du fonctionnement de l'outil qui leur garantissent l'acquisition d'une autonomie d'apprentissage.

Quelques questions générales auxquelles les participants devraient pouvoir répondre à l'issue de deux premières séances:

Que faut-il entendre par "traitement de l'information"?

Comment un ordinateur équipé de différents logiciels traite-t-il l'information?

Quelles nouvelles perspectives cela ouvre-t-il?

Quels nouveaux modes de travail cela induit-il?

(Etant donné la rapidité de l'évolution, tant du matériel que du logiciel...) Y a-t-il des invariants dans le domaine de l'informatique et des technologies de l'information qui garantissent des apprentissages utiles à long terme?

Quelques savoir-faire espérés:

Editer un dessin (le créer, le sauvegarder, en modifier le contenu,...) selon les deux modes principaux disponibles

Décrire, à tout moment du travail, sur quels supports se trouve un document (disque dur, disquette, écran, mémoire centrale...)

Décrire les applications présentes en mémoire à tout moment du travail et les rendre actives

PROGRAMME

La progression est dictée par le groupe, par l'intérêt qu'il marque pour l'un ou l'autre des thèmes abordés. L'objectif étant de rendre les utilisateurs aussi autonomes que possible, les matières abordées ne sont importantes que dans la mesure où elles ne mettent pas en péril l'acquisition de cette autonomie.

A propos d'exécutants et de "faire faire"...

- Une introduction très imagée du comportement attendu de l'utilisateur efficace et des problèmes qu'il devra résoudre: primitives, stratégies, traduction...
- Exercices de mise en condition

L'exécutant ordinateur

- Le principe de la numérisation (existence de codages), une source de réponses à de nombreuses questions et de solutions à de nombreux problèmes: exercice de codage d'un texte
- Quelques éléments indispensables de connaissance des ordinateurs en vue de conditionner de bons réflexes de la part de l'utilisateur et notamment:
 - un schéma simplifié et (assez) universel de l'ordinateur accompagné d'une évocation des différents périphériques
 - une description des différents supports d'information: papier, mémoire de masse, mémoire centrale...

L'exécutant "dessin par points"

- Comment est-il possible de numériser un dessin (exercice de codage)?
- Les conséquences sur les primitives attendues de ce type d'exécutant
- Problèmes à résoudre avec les exécutants de cette catégorie disponibles sur place

L'exécutant "dessin vectorisé"

- Comment éviter les inconvénients liés au codage du dessin par points?
- Les conséquences sur les primitives attendues
- Problèmes à résoudre avec les exécutants de cette catégorie disponibles sur place

La valse (ou la métamorphose) des exécutants: évolution au cours d'une séance de travail

- Description de l'évolution de l'état de la mémoire centrale au cours d'une séance de travail
- Lien avec les primitives du système d'exploitation qui assurent la communication entre la mémoire centrale et la mémoire de masse

1. A propos d'exécutants et de "faire faire"...

Depuis quelques décennies, on assiste à des modifications profondes et des extensions fulgurantes des pratiques en matière de traitement de l'information. Le mot utilisé jusqu'il y a peu pour désigner à la fois, la science, les techniques, les pratiques fondamentalement nouvelles a été le mot "**informatique**". Aujourd'hui, avec la banalisation de certaines de ces pratiques, le terme a plutôt tendance à se rattacher à nouveau à une activité professionnelle relativement précise, à un métier. Pour caractériser l'envahissement de l'électronique dans la vie de tous les jours, on parle plus volontiers de "**technologies de l'information**". Ces appellations et ces changements de vocabulaire sont loin d'être innocents. Le pluriel du mot "*technologies*" traduit une grande variété des domaines d'application et une impossibilité, même pour les spécialistes, d'assurer une expertise dans tous ces domaines. Il met en évidence l'aspect didactique que nécessite une bonne perception et une appréhension correcte du phénomène. Quant au mot "*information*", il apparaît maintenant comme l'élément central du problème. Une autre révolution relativement récente est liée à la banalisation des réseaux à tous les niveaux d'utilisation. Le moindre PC, même domestique, est périodiquement connecté au réseau des réseaux, c'est-à-dire *Internet*. Les possibilités d'exploitation sont nombreuses et les services disponibles se multiplient (courrier électronique, home banking, télé-achat,...). Pour cette raison, on parle aujourd'hui volontiers de "**technologies de l'information et de la communication**" (en abrégé *TIC*).

Il ne s'agit donc pas de prendre les choses à la légère. L'acquisition de savoir-faire et de compétences dans le traitement de l'information demande du temps, de la pratique, mais surtout de la clairvoyance.

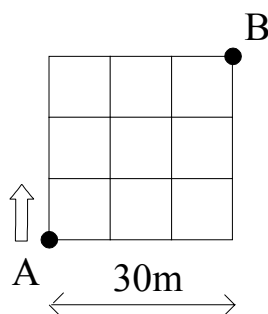
Dans cet ordre d'idée, nous avons l'ambition de faire de vous des utilisateurs efficaces, ou en tous cas, de vous montrer une voie qui vous conduira à le devenir. Nous insisterons donc souvent sur les aspects méthodologiques. Nous passerons fréquemment du temps à discuter ensemble et à réfléchir à des questions dont les réponses risquent fort de vous faire comprendre quels sont les enjeux de ce défi que vous vous lancez aujourd'hui et de vous montrer à quelles conditions vous pourrez le relever.

Mais bien sûr, compétence et expérience sont les produits d'un subtil mélange de théorie et de pratique. Si donc cette dernière faisait un peu défaut dans le cadre strict de la formation que vous suivez, il nous paraît important que vous la développiez entre les séances de formation avec l'aide et la complicité d'une personne qui sera à votre écoute (téléphonique ou électronique) selon des modalités qui vous sont précisées sur le site du [CeFIS](#).

En guise de mise en condition, nous allons vous proposer un petit problème qui va nous servir de point de départ à une première réflexion et qui donnera d'emblée un ton souvent adopté au début de cette formation.

Histoire d'extra-terrestre

Cette histoire est tirée de F. SASS, E. VANDEPUT *Informatique utile et raisonnée Tome I Notions et méthodes fondamentales* Van In Lier 1993



Un martien se trouve dans un parc dont les allées (symbolisées par des lignes) sont séparées de 10 mètres. Il se trouve en A, dans la direction indiquée par la flèche. Et vous devez lui donner les ordres pour qu'il se rende en B.

Les seules actions dont il est capable sont de se déplacer de 10 mètres d'un coup, droit devant lui, et de pivoter d'un quart de tour à gauche. Les choses se compliquent un peu si vous savez que pour lui faire accomplir

Un peu de vocabulaire...

Qu'il s'agisse d'un martien, d'un robot, ou d'un quelconque personnage ou dispositif auquel on peut donner des ordres ou des commandes, nous parlerons d'un **exécutant**.

Lorsque nous désignerons des actions dont est capable un exécutant limité, nous parlerons d'**actions primitives**.

Diriger un exécutant, c'est en quelque sorte définir une stratégie pour lui faire accomplir une tâche donnée, simple ou complexe, à partir de ses primitives.

Ainsi, la conduite du martien de A en B nécessite une stratégie, une route à suivre qui va de temps en temps se heurter à des difficultés liées à l'absence de certaines primitives (pour *tourner à droite*, par exemple). Dans ces cas-là, il s'agit d'examiner si, en décomposant la difficulté, les primitives existantes permettent de la lever (*tourner à gauche* trois fois de suite équivaut à *tourner à droite*).

Si pour solutionner des problèmes, une connaissance minimale de l'exécutant est nécessaire, la résolution de problèmes (en termes d'exécution de tâche) est aussi un moyen de découvrir un peu plus de quoi il est capable. S'installe donc une dynamique d'auto-apprentissage qui peut s'avérer extrêmement efficace au travers de la succession des différents problèmes rencontrés dans le travail de tous les jours.

2. L'exécutant ordinateur

La réflexion qui précède n'a d'autre objet que de vous faire prendre conscience de la manière dont vous serez obligés de vous comporter lorsque vous voudrez effectuer un travail quelconque au moyen d'un ordinateur. Il vaudrait d'ailleurs beaucoup mieux parler de "*faire exécuter*", même si vous avez tout de même le sentiment tout à fait justifié de travailler vous aussi.

En réalité, **il convient de distinguer votre travail qui consiste à guider un exécutant, du travail de cet exécutant. En réalité, c'est lui qui traite les informations que vous lui fournirez, de la manière dont vous lui indiquerez.**

Il importe donc que vous fassiez plus ample connaissance avec cet exécutant, et surtout, que vous preniez conscience des éléments fondamentaux de sa conception qui orienteront votre façon de travailler avec lui.

A. L'ordinateur n'est rien sans un programme pour le gouverner

Les véritables exécutants auxquels vous aurez à faire répondent tous à l'équation suivante:

$$EXECUTANT = ORDINATEUR + PROGRAMME$$

Qu'est-ce qu'un programme? C'est, d'une certaine manière, l'ensemble des capacités de l'exécutant. Ainsi, si le martien existe "en dur", il existe aussi parce qu'il comprend certains ordres. C'est le programme du martien qui le fait avancer de 10 mètres lorsque le mot *blanc* est prononcé.

Le mot *logiciel* est souvent utilisé à la place du mot *programme*. On distingue aussi le *hardware* du *software*, le *matériel* du *logiciel*.

Pour être utilisable, un ordinateur doit donc posséder dans sa "mémoire", un programme qui lui confère différentes capacités. Vous pouvez vous figurer qu'il existe, par exemple, des exécutants "traitement de texte". N'importe quel ordinateur dont la mémoire contient un programme le transformant en

machine à travailler les textes, une sorte de machine à écrire évoluée (quoique, cette image n'est peut-être pas la meilleure) sera pour nous un exécutant "traitement de texte". Nous parlerons aussi d'exécutants "dessinateurs", pour désigner tout ordinateur dont la mémoire contient un programme de dessin, d'exécutants "comptables" etc. En réalité, chaque programme transforme l'ordinateur en un exécutant capable d'effectuer un certain nombre d'actions répondant à un type de travail donné.

Il existe donc des millions d'exécutants, compte tenu de la très grande diversité des programmes existants et des ordinateurs les supportant. Vous comprenez qu'il vous faut, dès maintenant, vous faire une raison de ce grand nombre d'exécutants présents, et surtout à venir.

Les choses se compliquent lorsqu'on se rend compte que le programme actif peut changer en cours de travail (l'exécutant mue) et même, que plusieurs programmes peuvent être actifs simultanément. Mais pour ce qui nous concerne, nous ne dialoguerons jamais qu'avec un seul exécutant à la fois. L'essentiel est de savoir lequel.

Pour vous permettre de vous familiariser avec eux, nous allons commencer par vous donner une idée de la manière dont est conçu un ordinateur, comment très globalement il fonctionne, et surtout, quelles en sont les limites bien réelles (certains préjugés pourraient nous laisser croire le contraire).

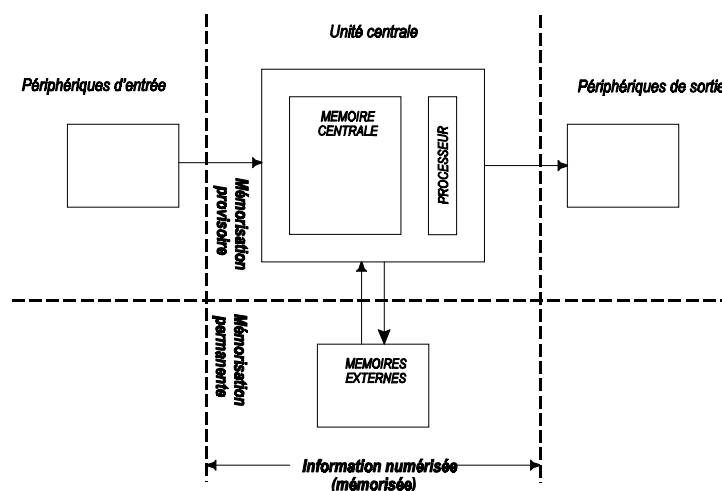
B. Grâce à ses programmes, l'ordinateur traite des données

Un exécutant est généralement spécialisé dans une tâche particulière (traitement de texte, dessin, navigation sur Internet,...). De ce fait, il manipule des données (textes, dessins, pages Web,...).

Et donc, après avoir distingué le matériel du logiciel, nous distinguerons encore l'information qui est produite et traitée par les différents exécutants issus des nombreux couples possibles *ordinateur + logiciel*. Nous donnerons souvent à ce type d'information le nom de *données* pour la démarquer des *programmes* ou logiciels qui dans notre conception imagée font partie intégrante des exécutants.

C. Le schéma simplifié d'un ordinateur (du point de vue de son fonctionnement)

Nous avons bien écrit *schéma* et non *dessin*. Cette remarque est importante car elle est à la base de confusions gênantes chez de nombreux utilisateurs. Ces confusions sont de nature à rendre précaire, voire inefficace tout travail qu'il font avec un ordinateur. Effectivement, il y a ce que l'on voit et ce que l'on imagine. En matière de fonctionnement d'un ordinateur, c'est ce qu'on imagine qui est primordial.



On a souvent dit de l'ordinateur qu'il sert à traiter l'information. Cette petite phrase appelle de nombreux commentaires. A commencer par: quelle signification faut-il donner à ce mot *information*? quels sont les traitements possibles et pourquoi?...

Nous pourrions dire que depuis qu'ils existent, tous les ordinateurs répondent au schéma simplifié qui précède.

Il convient de mettre ce schéma en accord avec ce que vous pouvez observer lorsque vous vous trouvez devant un ordinateur.

Comme vous pouvez le constater, en amont et aval du traitement de l'information par l'ordinateur se trouvent des unités d'entrée et de sortie. Ces unités, vous les trouvez généralement autour de la partie centrale de l'ordinateur. Pour cette raison, on leur donne souvent le nom de *périphériques* (d'entrée et de sortie). En guise de point de départ, pourriez-vous citer des éléments, des parties de l'ordinateur qui vous paraissent être des périphériques en précisant de quelle nature ils sont (entrée, sortie ou les deux)?

Périphériques d'entrée:

.....

.....

.....

.....

.....

.....

Périphériques de sortie:

.....

.....

.....

.....

.....

.....

Périphériques d'entrée/sortie

.....

.....

.....

.....

.....

.....

Une question fondamentale se pose à leur sujet: à quoi servent-ils?

Pour répondre à cette question, il faut en savoir un peu plus sur la composition de la partie centrale de l'ordinateur, l'unité centrale de traitement (en anglais *CPU: Central Processing Unit*).

Ce n'est un secret pour personne, l'ordinateur fonctionne grâce à des circuits électroniques. Depuis le début, c'est l'électricité qui est à la base de tous les développements de l'informatique. Les conséquences en sont énormes et conditionnent depuis toujours la nature des produits informatiques et leur influence sur le mode de travail des utilisateurs.

La partie centrale d'un ordinateur se compose d'une mémoire (centrale). Cette mémoire est destinée à contenir les programmes qui transforment l'ordinateur en différentes sortes d'exécutants et les informations que ceux-ci vont devoir traiter. Elle est composée de circuits électroniques. La caractéristique de chacun de ces circuits, c'est qu'à tout moment, il n'a que deux états possibles: ou bien il est parcouru par un courant, ou bien il ne l'est pas. Voilà sous quelle forme se trouve l'information dans l'unité centrale d'un ordinateur.

Cela implique qu'**une transformation de l'information est nécessaire en amont comme en aval**. Dans les deux cas, cette transformation se justifie par le fait que l'information telle que nous avons l'habitude de la traiter n'a pas la forme de milliards de circuits sous tension ou non. **Ces transformations se font de manière transparente pour l'utilisateur par l'intermédiaire des périphériques** dont nous avons déjà parlé. Toutefois, cela n'est pas sans implications. Nous allons d'ailleurs nous y intéresser sérieusement.

Pour l'instant, imaginez-vous que dans la mémoire centrale de l'ordinateur, toute information est irrémédiablement convertie en une suite composée des deux chiffres 0 et 1, une manière commode de symboliser cette présence ou cette absence de courant dans un circuit.

Le rôle des périphériques d'entrée est de permettre une numérisation de l'information. Monsieur de La Palisse l'aurait bien dit: **pour pouvoir être traitée par un ordinateur, une information doit pouvoir être numérisée**. Celui des unités de sortie est de redonner à l'information un aspect familier à l'être humain. Celui des unités d'entrée/sortie, c'est de lui permettre d'être fixées sur un support qui en assure réutilisation.

Le parcours de l'information est donc linéaire. Elle est numérisée (codée) avant d'être traitée puis restituée grâce à un décodage qui lui redonne l'aspect original. On comprend dès lors qu'il faille un **coordonnateur de toutes ces opérations**. Ce rôle est assuré par **le processeur**, un autre composant électronique capable de réaliser un nombre limité d'opérations élémentaires en des temps records.

Quelques processeurs (marque et type)

.....

.....

.....

.....

.....

.....

D. Les mémoires

De ce qui précède, vous retiendrez que le passage en mémoire centrale de l'information est limité dans le temps. La mémoire centrale est une mémoire de travail. Une fois traitée, l'information résultante doit être copiée sur un support pour qu'une trace puisse en être conservée. **La mise hors tension de l'ordinateur a pour conséquence de vider la mémoire centrale de tout son contenu.** Quels sont les supports possibles?

Il y a d'abord le support traditionnel que constitue le papier. Si ce support est quelque peu durable, il est difficilement réutilisable car le texte n'est plus sous une forme aisément numérisable. Nous allons examiner de suite les possibilités de numérisation.

Il y a aussi et surtout les supports magnétiques tels les bandes enregistreuses, les disquettes et disques durs et des supports plus modernes comme les cédéroms ou le vidéo-disque (bien qu'actuellement, avec ces derniers, le transfert d'information se fasse essentiellement en entrée).

Observez qu'on parle d'*entrée* en ce sens que ce type de support permet d'entrer de l'information dans la mémoire centrale (lecture ou chargement). On parle aussi de *sortie* en ce sens que certains supports (un disque dur, par exemple) autorise la sortie de l'information de la mémoire centrale vers eux-mêmes (écriture ou enregistrement). A noter qu'*entrer* et *sortir* ont ici la signification très particulière de "*faire une copie de ... dans ou sur ...*".

On peut admettre que sur ces différents supports, l'information garde une forme numérisée, ce qui en facilite la réutilisation par un ordinateur. C'est la raison pour laquelle ils portent également le nom de *mémoires* mais on parlera de *mémoires externes* ou de *mémoires de masse* pour les désigner et pour les distinguer de la mémoire centrale. Contrairement à cette dernière, **les mémoires externes gardent leur contenu si l'ordinateur n'est plus sous tension.**

La taille des mémoires externes et centrale se mesure en fonction de la quantité d'information qu'elles peuvent stocker. L'unité de mesure est directement liée à la manière dont celle-ci est codée. Nous pourrions donc en parler plus à l'aise après avoir répondu à la question: comment numériser l'information?

E. La numérisation

L'histoire de l'informatique et le développement des technologies sont construits sur le même objectif qui est de rendre de plus en plus d'informations numérisables. A l'origine, c'est essentiellement pour calculer que les ordinateurs étaient construits. Les problèmes de numérisation étaient donc limités dans la mesure où il suffisait de convertir les nombres du système décimal dans le système binaire (seulement deux symboles: 0 et 1) et inversement. Mais assez rapidement, on s'est rendu compte que les caractères pouvaient aussi être numérisés, ce qui ouvrait la porte à la création de nouveaux programmes et donc de nouveaux outils.

Imaginez une manière de coder les lettres de l'alphabet qui vous permettent de communiquer avec un correspondant en ne lui transmettant que des chiffres (de 0 à 9)

.....

.....

.....

Le nombre de caractères différents utilisés pour écrire des textes n'est pas infini. Bien sûr, les lettres majuscules et minuscules représentent des caractères différents auxquels il faut ajouter les chiffres, les signes de ponctuation et, particularités de certaines langues (latines, scandinaves,...) les caractères accentués. Le tout ne dépasse guère la centaine.

Un seul circuit électronique permettant de coder deux caractères (puisque deux symboles sont disponibles), deux circuits permettent d'en coder quatre, trois circuits huit et ainsi de suite, huit circuits donnent accès à 256 codes différents. On pourrait s'étonner de ce choix de huit (alors que sept auraient pu suffire) mais les problèmes de conversion étant ce qu'ils sont, il est beaucoup plus simple de passer d'un système octal (à huit symboles) à un système binaire et réciproquement. Les circuits ont donc été rassemblés par paquets de huit, ce qui a eu comme conséquence de faciliter le codage des caractères habituels et de quelques autres puisque le nombre de possibilités dépassait les nécessités.

Ainsi, selon un code aujourd'hui standardisé, la lettre A (majuscule) est codée 01000001 dans la mémoire d'un ordinateur, la lettre B (majuscule) 01000010 et ainsi de suite...

Un circuit correspondant à un chiffre binaire porte le nom de **bit** (contraction de l'expression anglaise "binary digit": chiffre binaire). **La place que nécessite le codage d'un caractère est appelée octet** (un octet équivaut donc à huit bits). La taille des mémoires de masse comme de la mémoire centrale d'un ordinateur se mesure donc en *octets* (en anglais *bytes*). Une évolution rapide des technologies liées au stockage de l'information fait qu'aujourd'hui, on mesure davantage en *kilo-octets* (Ko ou Kb), en *méga-octets* (Mo ou Mb) voire en *giga-octets* (Go ou Gb). Les préfixes *kilo*, *méga* et *giga* correspondent habituellement à des puissances de dix (10^3 , 10^6 et 10^9). Dans ce contexte, ils représentent les puissances de deux (système binaire) les plus proches de puissances de dix soit pour l'anecdote: 1.024, 1.048.576 et 1.099.511.627.776 octets (pour rappel 1 octet / 1 caractère).

Notez ici quelques tailles actuellement courantes pour la mémoire centrale d'un ordinateur, une disquette, un disque dur, un cédérom, un vidéodisque:

.....
.....
.....
.....

F. Les conséquences (dramatiques) de la numérisation

Si l'on vous fournit un texte écrit en polonais (à supposer, et c'est fort probable, que vous ne connaissiez pas le polonais), serez-vous capable de compter le nombre de mots du texte? Pourrez-vous souligner tous les adjectifs, les mots de trois lettres, les verbes conjugués? Pourrez-vous y retrouver un mot donné? Si on vous donne un dictionnaire polonais-italien, pourrez-vous traduire ce texte en italien?

Quelles sont vos réponses à ces questions et pourquoi?

.....
.....
.....

G. La valse des exécutants

Vous êtes presque parés pour une confrontation avec un premier exécutant. Il nous reste à éclaircir un point important: que se passe-t-il à la mise sous tension d'un ordinateur? d'où viennent les exécutants? comment évoluent-ils au cours d'une session de travail?

Il y a une espèce de petite contradiction dans ce que nous avons affirmé jusqu'à présent. Si nous nous retrouvons, à un moment donné, devant un exécutant, c'est qu'un programme se trouve dans la mémoire centrale. Un exécutant est en effet la combinaison d'un ordinateur et d'un programme. On comprend bien qu'un tel exécutant puisse transformer ses capacités s'il peut faire entrer d'autres informations dans la mémoire. Ainsi, un exécutant peut évoluer au cours d'une session de travail ou laisser la place à un autre selon l'image que vous préférez. Mais nous avons affirmé qu'à la mise hors tension de l'ordinateur, la mémoire centrale de l'ordinateur se vidait complètement. Comment un programme peut-il donc se retrouver en mémoire si l'exécutant au départ a un "encéphalogramme plat"?

La réponse à cette question, c'est que la mémoire centrale d'un ordinateur n'est jamais complètement vide. Une petite partie de celle-ci (qui était d'ailleurs plus importante dans un passé récent) contient un "souffle d'intelligence" sous la forme de quelques instructions systématiquement exécutées à la mise sous tension de l'ordinateur. Cette partie ineffaçable de la mémoire centrale est appelée **mémoire morte** (ou *ROM, Read Only Memory*) par opposition à la **mémoire vive** (ou *RAM, Random Access Memory*). Les instructions qu'elle contient font que l'ordinateur n'est pas seulement un "tas de ferraille et de plastique" mais un exécutant très limité qui exécute, à chaque mise sous tension, le même travail. En gros, et pour faire simple, ce travail consiste à mettre en mémoire, à partir d'une mémoire externe (souvent le disque dur) des capacités minimales sous forme d'un programme ou groupe de programmes appelé **système d'exploitation**.

Une fois le *système d'exploitation* en mémoire, l'exécutant de départ a changé. On est maintenant en présence d'un exécutant capable d'**exploiter** ses ressources et notamment, ses périphériques et sa mémoire centrale.

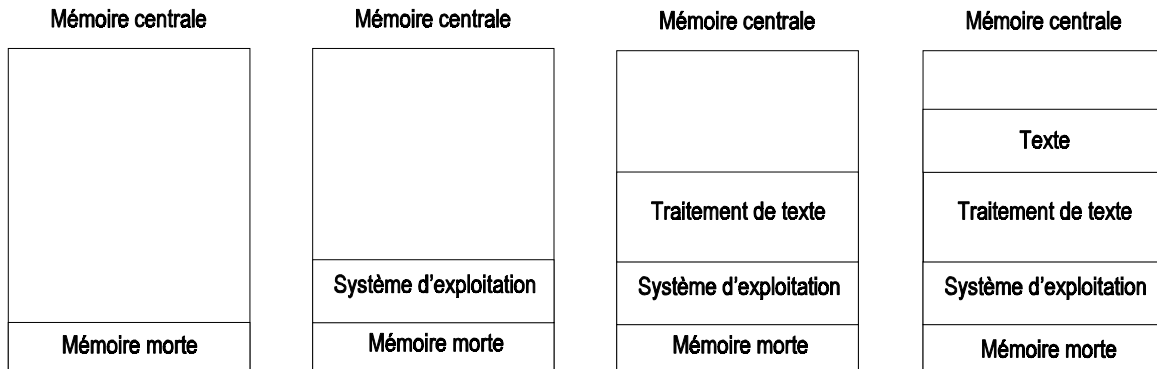
A partir de ce moment, l'utilisateur que vous êtes peut prendre en charge cet exécutant pour lui confier des tâches telles la gestion des supports d'information (effacer, faire des copies, déplacer...) et surtout le chargement en mémoire d'autres programmes qui vont encore affiner les qualités de l'exécutant comme, par exemple, mettre en mémoire un programme de traitement de texte.

Comme nous l'avons laissé supposer, l'exécutant *système d'exploitation* considérera que l'information peut être de deux types suivant qu'elle correspond à des programmes ou à des données. Néanmoins, pour ce qui est de la gestion globale de ces deux types d'information (copie, destruction, déplacement,...) ils seront traités de la même manière. Une des façons de s'en rendre compte est de constater que tant les programmes que les données sont encapsulés dans ce que l'on appelle des **fichiers**.

Décrivons un exemple de session de travail.

A la mise sous tension de l'ordinateur, le processeur exécute les quelques instructions contenues dans la mémoire morte de l'ordinateur. Parmi celles-ci, des instructions ordonnent la recherche dans les mémoires externes (disquette et à défaut disque dur) du système d'exploitation pour en faire une copie en mémoire. A la fin des opérations, l'utilisateur que vous êtes n'est, en principe, pas intervenu et il est en présence d'un exécutant "système d'exploitation" ou, si vous voulez, un exécutant de base.

Vous pouvez alors commander à cet exécutant de charger dans sa mémoire d'autres capacités sous forme d'un programme supplémentaire, un programme de traitement de texte, par exemple. Après cette opération, vous êtes devant un exécutant "traitement de texte". Attention, l'expression **charger en mémoire** a le sens précis de **faire une copie dans la mémoire centrale d'informations se trouvant dans une mémoire externe**.



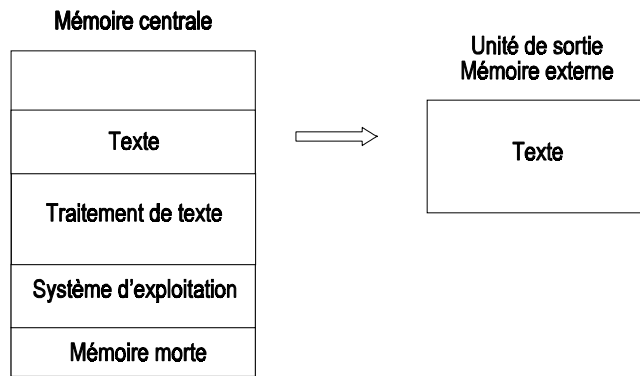
Cet exécutant, vous pouvez le faire travailler dans le sens où vous lui fournissez des informations, disons des données, sous forme d'un texte que vous encodez au clavier. Ces données se retrouvent également dans la mémoire centrale.

Conséquence: **dans la mémoire centrale**, se retrouvent aussi bien des **programmes** qui confèrent à l'exécutant une intelligence très artificielle (pensez au côté formel des traitements réalisables) que des **données**: un texte, un dessin, un air de musique... et toute information également numérisable. Il en sera donc de même pour les mémoires externes: un disque dur, une disquette, un cédérom... Les uns comme les autres contiendront deux types d'informations: des programmes et des données. C'est au système d'exploitation de gérer correctement cette cohabitation.

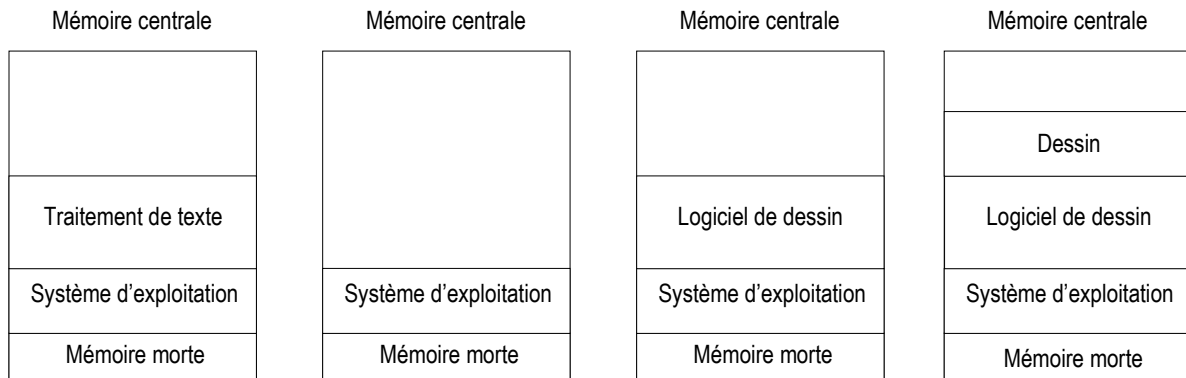
La suite des opérations peut globalement consister en deux choses: vous souhaitez travailler avec un autre exécutant ou vous souhaitez cesser le travail. Dans l'un comme dans l'autre cas, des précautions sont à prendre.

Pour travailler avec un autre exécutant, il faut que ses "capacités" soient présentes en mémoire sous forme d'un autre programme, un logiciel de dessin, par exemple. Si vous ne souhaitez plus travailler avec l'exécutant précédent, une bonne chose est de le faire disparaître de la mémoire centrale, lui et le travail qu'il a permis de faire. C'est l'exécutant "système d'exploitation" qui va prendre cette tâche en charge. Rappelez-vous que la mémoire centrale est une mémoire de travail et que les informations qu'elle contient sont de toute façon vouées à disparaître lors de la mise hors tension de l'ordinateur. La disparition du programme ne pose pas de problème, puisqu'il s'agit d'une copie. Celle des données est plus gênante, car aucune copie de ces données n'existe sur un support non "volatile". Vous devez alors faire exécuter par le système d'exploitation une opération de sauvegarde des données en mémoire externe (le plus souvent disquette ou disque dur) avant de lui donner l'ordre de vider les données, voire le programme de la mémoire centrale.

La sauvegarde effectuée, il faudra demander au système d'exploitation de décharger le texte (les données) et le traitement de texte (le programme) avant de faire procéder au chargement d'un autre



programme. Qui dit nouveau programme dit nouvel exécutant donc autre type de tâche (par exemple, du dessin).



Quelques commentaires importants sont à faire. D’abord, la sauvegarde et le déchargement du texte et du traitement de texte ne sont pas obligatoires pour pouvoir travailler avec un autre exécutant. Il est possible de mettre l’exécutant “traitement de texte” en sommeil et de le rappeler plus tard. Toutefois, si une tâche est vraiment terminée, c’est une bonne habitude que de la clôturer de la manière que nous venons de décrire.

Il faut également dire un mot de l’exécutant “système d’exploitation”. Comme le montrent les schémas qui précèdent, dès qu’il est en mémoire, il y reste jusqu’à la fin de la session de travail. C’est assez logique dans la mesure où les fonctionnalités de base doivent rester disponibles.

Si comme nous l’avons dit, la vocation d’un système d’exploitation est essentiellement de permettre d’exploiter les ressources, il faut préciser qu’aujourd’hui, on lui adjoint souvent des fonctionnalités liées à la gestion d’un environnement graphique. Cet environnement assure un confort d’utilisation dans la mesure où de nombreuses manipulations peuvent être réalisées de manière dynamique (en faisant glisser la souris, par exemple). Quelquefois, le programme de gestion de l’environnement graphique est séparé du système d’exploitation (*Windows 3.0, 3.1, 3.11*) quelquefois, il y est intégré (*Windows 95* et versions ultérieures). Plus rarement, on travaille sans environnement graphique avec un système d’exploitation fonctionnant en mode texte (*MS-DOS*).

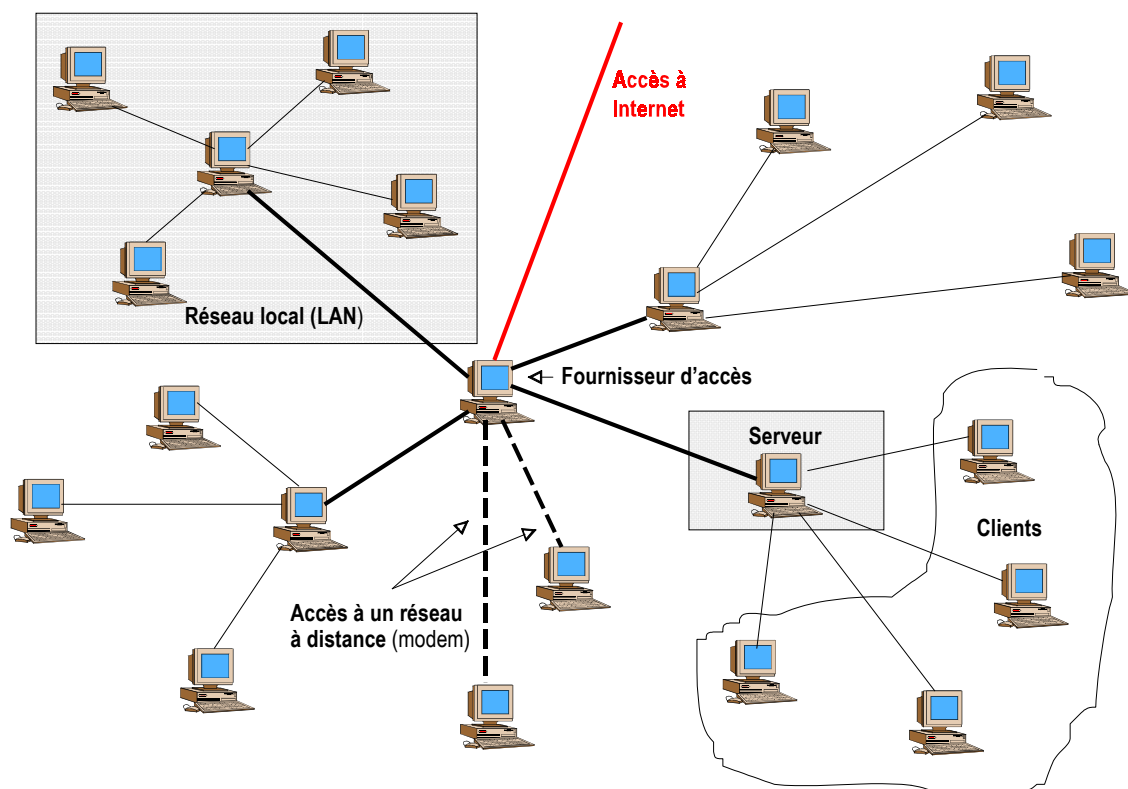
H. Vers un schéma plus général

Tout un volet des technologies de l'information n'a pas encore été abordé dans notre description de l'ordinateur. Il s'agit de la communication. Si la plupart des ordinateurs personnels des années 80 étaient parfaitement isolés, ceux de la fin des années 90 et du XXI^{ème} siècle sont presque tous reliés à d'autres ordinateurs par le biais d'un réseau local et/ou d'un réseau à distance grâce, par exemple, à un simple fil téléphonique ou à un câble de télédistribution.

Cet état des choses n'est pas sans conséquences. La quantité d'information disponible a connu un développement exponentiel. De nombreux services demandant hier déplacement et perte de temps sont disponibles à portée de souris. En revanche, d'autres phénomènes voient le jour. Les virus, ne se communiquant jusque là que par l'échange de disquettes, se propagent aujourd'hui le long des lignes de communication, souvent à l'insu des responsables de leur envoi. Nous aurons l'occasion de traiter des avantages et des inconvénients liés à la prolifération des réseaux lors de séances ultérieures. Pour l'instant, contentons-nous d'étoffer notre schéma descriptif de l'ordinateur en le plaçant dans le contexte de cette communication à outrance.

Ce qui est essentiel pour la compréhension, c'est d'admettre que l'information se déplace ou se reproduit dans des mémoires de différentes sortes. Prenons un exemple simple: l'enregistrement d'un texte consiste en la copie du texte de la mémoire centrale vers un disque dur ou une disquette. Il en est de même pour les activités qui font appel aux réseaux. L'information s'échange (envoi de copies) entre les mémoires de différents ordinateurs.

Un premier schéma nous donne un aperçu de la manière dont un ordinateur peut être connecté à quelques, voire à des millions d'ordinateurs.



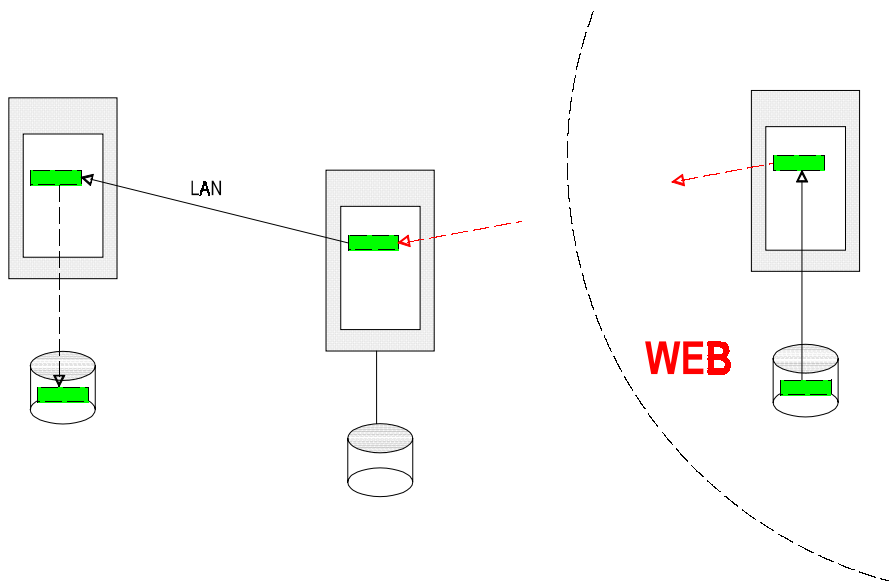
Il arrive fréquemment que plusieurs ordinateurs d'un même établissement, d'une même entreprise ou administration soient reliés entre eux dans ce que l'on appelle un réseau local (en anglais *LAN* comme *Local Area Network*). C'est le cas des ordinateurs des centres cybermédiés installés dans les écoles. Dans le schéma le plus courant et le plus apprécié aujourd'hui, ils sont reliés à un ordinateur généralement plus puissant qui leur rend des services que l'on appelle pour cette raison, un **serveur**. Les autres sont donc considérés comme des clients.

Bien entendu, n'importe quel ordinateur est un serveur dès qu'il possède, dans sa mémoire centrale, un programme qui lui permet de rendre au moins un service aux autres. Il est donc possible qu'un ordinateur serveur pour un service donné soit un client pour un autre service. Par exemple, Jo, le serveur du bar "A la bonne chope", est client chez le coiffeur "Cauptif" (et le coiffeur se fait servir à boire par Jo), mais cette situation n'est pas la plus courante. On parle donc fréquemment d'une répartition des ordinateurs et des services dans une **architecture client-serveur**.

Notez que la liaison à un réseau peut se faire à distance, via un modem (ligne téléphonique, câble de télédistribution,...) et être occasionnelle (pointillés dans le schéma).

Dans un réseau local, un ou plusieurs ordinateurs peuvent rendre un service d'accès à Internet. Les différents LAN sont reliés les uns aux autres par des lignes généralement à plus gros débit (comme les petites rues de deux villages sont reliées par une route plus importante). Et ainsi de suite, les routes entre villages finissant par mener aux villes et aux autoroutes. Finalement, c'est une véritable toile (Web) qui est constituée.

Tout ceci ne nous dit pas comment les ordinateurs communiquent entre eux et donc, comment ils s'échangent de l'information. Si le processus est complexe dans ses détails, la seule chose qu'il nous importe de comprendre, c'est la localisation des informations (quel ordinateur, mais aussi quelle mémoire). Nous détaillons encore le schéma sur base d'un exemple, celui de la visualisation d'une page d'un site web. Il s'agit d'un schéma dans lequel nous représentons les mémoires les plus importantes de l'ordinateur: sa mémoire centrale (un rectangle blanc) et ses mémoires de masse (symbolisées ici par des cylindres).



Voici un exemple, sur lequel nous reviendrons plus en détail lorsque nous parlerons de navigation sur Internet.

Imaginez qu'au gré de votre navigation de site en site, vous découvriez l'image d'un paysage de rêve, et il vous intéresse d'en disposer sur votre ordinateur. Vous répondrez naturellement que c'est déjà fait puisque vous êtes en train de visionner cette image. Si vous avez bien intégré ce qui précède, vous aurez compris que la représentation numérique de cette image est présente dans la mémoire centrale de votre ordinateur. Si vous souhaitez la conserver à long terme, il s'agira de l'enregistrer, par exemple sur le disque dur de cet ordinateur. Les manipulations à réaliser sont relativement simples et seront décrites plus tard.

Il reste une question importante: comment cette image lointaine est-elle parvenue dans la mémoire de votre ordinateur? Le processus sera étudié en détail mais ce qu'il est bon d'en retenir pour l'instant, c'est qu'un simple clic de souris (celui qui a provoqué l'affichage de la page sur laquelle cette image se trouve) a été interprété comme une commande, à l'adresse d'un ordinateur serveur sur le Web, pour envoyer les données en question. L'ordinateur serveur a donc effectué une copie des données (entre autres, l'image) de sa mémoire externe dans sa mémoire centrale avant d'envoyer cette copie sur le réseau à destination de la mémoire centrale de votre ordinateur. C'est à vous de décider si vous souhaitez enregistrer ou non cette copie.

Cette description est évidemment simpliste. Nous n'avons pas répondu à toutes les questions comme, par exemple, comment votre ordinateur sait-il à quel ordinateur il doit s'adresser pour obtenir la page souhaitée? Mais les réponses viendront en temps utile. Ce qui compte pour l'instant, c'est de comprendre que le traitement de l'information est fréquemment un jeu de copie d'information d'une mémoire vers une autre du même type ou d'un type différent. Vous devrez donc avoir une assez bonne idée de la localisation provisoire ou (temporairement) définitive de l'information que vous traitez.

I. Se connecter à un réseau

La connexion à un réseau comporte un certain risque, tant pour l'utilisateur, que pour celui qui met des ressources à disposition. Lorsque l'utilisateur télécharge un fichier, il n'est pas exempt de recevoir un virus en prime. De même, celui qui met son ordinateur au service des autres, ne souhaite pas que ces derniers viennent mettre à sac le contenu de ses disques durs.

Pour ces raisons notamment, toute connexion à un réseau, que ce soit localement ou à distance, nécessite une autorisation. Cette autorisation est assez classique et ressemble un peu à la démarche que vous effectuez lorsque vous retirez de l'argent dans un distributeur automatique de billets. Vous vous identifiez (en introduisant votre carte dans le distributeur), puis vous fournissez un mot de passe (votre code) pour accéder aux opérations. Lorsque vous vous connecterez (et si vous travaillez dans un réseau local, ce sera dès l'ouverture d'une session de travail), vous devrez fournir une identification (un *login*), puis le mot de passe qui lui correspond. Si votre *login* n'a rien de secret, votre mot de passe ne doit en aucun cas être divulgué. Lorsque vous l'encodez au clavier, ce sont des étoiles qui apparaissent à l'écran.

Encore une chose, si lorsqu'on vous demande un *login* et un mot de passe, vous annulez l'opération, vous pourrez travailler mais vous n'aurez pas accès aux ressources du réseau. Si, pour des raisons de facilité, il n'y a pas de mot de passe à fournir, cela ne vous dispense pas de fournir votre *login*

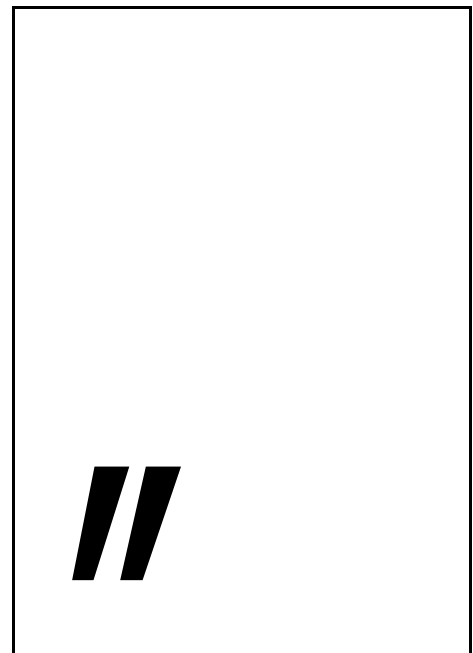
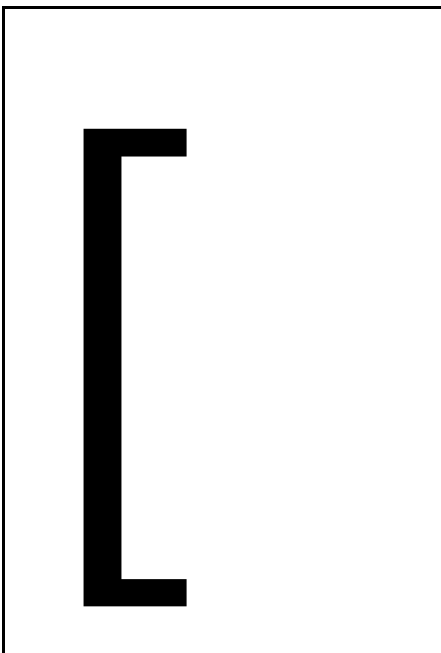
3. L'exécutant "dessin par points"

A. Comment numériser une image?

Vous avez imaginé comment numériser des caractères pour faire en sorte qu'un ordinateur soit à même de traiter des textes. Posez-vous maintenant la même question à propos d'une image.

Vous désirez communiquer des images à un correspondant, mais vous devez le faire par l'intermédiaire d'un dispositif qui n'accepte que des chiffres. Comment pourriez-vous procéder?

A titre d'exercice, observez ci-après deux images très simples, représentées sur une feuille de papier. Pouvez-vous imaginer une technique, un dispositif qui permette de convertir ces images en une suite de chiffres que quelqu'un d'autre pourra reconvertir pour reconstituer les images?



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ne tournez pas la page si vous désirez rechercher vous-mêmes une solution.

choisissiez la couleur rouge dans la palette de couleurs, que vous choisissiez un outil vous permettant de dessiner des cercles (l'ordre de ces deux opérations a relativement peu d'importance), puis que vous définissiez, au moyen de la souris, une zone d'une certaine forme dont les pixels auront cette couleur.

Si par la suite, vous décidez de dessiner un rectangle vert qui recouvre totalement ce cercle rouge, c'est un peu comme si vous utilisiez de la peinture, les pixels de couleur rouge ont pris la couleur verte. Le cercle, comme le rectangle n'existent pas en tant que formes géométriques, mais comme ensembles de pixels.

Formes et traits

Les outils de dessin disponibles sont généralement le cercle ou l'ellipse, le rectangle, la ligne droite, la ligne brisée, la ligne courbe... Pour chacun d'eux, il importe de faire **préalablement** des choix comme: la couleur, l'épaisseur et le style du trait, la couleur ou le motif de remplissage...

La stratégie de dessin d'une forme ou d'un trait est donc la suivante:

- choisir l'outil qui correspond à la forme ou au trait
- choisir les attributs de la forme ou du trait
- définir la zone (au moyen de la souris)

Existe-t-il d'autres outils? Si oui, ils doivent procéder de la même manière à savoir, modifier la couleur des pixels d'une zone définie par l'utilisateur. C'est le cas de la **gomme**, un outil dont la taille est un attribut à fixer préalablement. Le principe est que les pixels situés dans les zones parcourues par la gomme voient leur couleur modifiée. On peut encore citer le pinceau, le crayon, l'aérographe,... qui sont des outils possédant comme les autres, plusieurs attributs à déterminer avant toute utilisation.

Remplissage

Nous avons insisté sur la nécessité de fixer les attributs d'un trait ou d'une forme avant de les dessiner. Il existe une possibilité de corriger un oubli, c'est d'utiliser avec précaution l'outil de remplissage. Celui-ci entraîne la modification de la couleur de tous les pixels adjacents de même couleur que celui qui est pointé avec la souris. Attention, il arrive qu'une image se colorie de manière uniforme après l'utilisation de cet outil!

Texte

Il existe un outil un peu plus particulier qui est l'outil texte. Il est vrai qu'une image, une affiche, un dessin sont souvent agrémentés d'un bout de texte. L'outil texte permet d'éditer du texte dans un cadre, c'est-à-dire de l'encoder et d'en modifier le contenu et la forme. Toutefois, lorsque cette édition est terminée, le texte est devenu un ensemble de pixels. Il n'est donc plus possible de l'éditer à nouveau. Il faut utiliser la gomme pour l'effacer.

Comment réparer les gaffes?

Il existe plusieurs manières de corriger des erreurs dues à la maladresse (manque d'habileté dans l'utilisation de la souris), à de mauvais choix d'attributs, à de mauvaises appréciations. Nous avons déjà parlé de la gomme qui est loin de résoudre tous les problèmes. Il existe un moyen plus radical. Puisque l'ordinateur mémorise (même temporairement dans la mémoire centrale), l'exécutant est de plus en plus souvent capable de mémoriser les dernières actions qu'il a effectué pour vous. De la sorte, il lui est donné de pouvoir reproduire la (les) situation(s) antérieure(s). Le vocable utilisé est **Défaire** ou **Annuler**. Inutile de dire qu'un utilisateur patenté use et abuse de cette opportunité qui n'est d'ailleurs pas l'apanage des logiciels de dessin.

S'il s'agit de petites retouches à faire au dessin, il existe une autre façon de procéder, propre aux logiciels de dessin par points, c'est d'employer le **zoom** pour modifier l'image pixel par pixel. L'image étant un ensemble de pixels, il est normal que la plus petite partie du dessin sur laquelle il soit possible de travailler soit le pixel.

Sélectionner, couper, copier, coller

Vous vous rendez compte que ce type d'exécutant va exiger que vous soyez assez sûrs de vous quant à la position des formes et des traits et aux attributs que vous leur donnez. Malgré tout, il vous sera possible de "faire bouger" certaines choses dans une certaine limite, grâce à un autre principe qui n'est pas non plus propre aux logiciels de dessin mais plutôt au système d'exploitation qui est toujours en arrière-plan. La technique du **sélectionner-couper-coller** ou du **sélectionner-copier-coller** est inspirée de ce que vous pourriez faire pour créer une affiche: découper des parties de dessin ou les polycopier et les assembler à votre guise. L'élément virtuel qui permet à nos exécutants de travailler de la sorte s'appelle le **presse-papiers**. Il vous est loisible de sélectionner une partie du dessin, de l'envoyer dans le presse-papiers puis de le récupérer. Seul ennui, dans les versions les plus anciennes des systèmes d'exploitation, le presse-papiers ne peut contenir qu'une seule sélection à la fois.

L'opération qui consiste à envoyer la sélection dans le presse-papiers en la faisant disparaître s'appelle **couper**. La même opération sans faire disparaître la sélection s'appelle **copier**. La récupération du contenu du presse-papiers s'appelle **coller**.

La sélection, avec un logiciel de dessin par points, se fait au moyen de la souris, soit à main libre, soit en zone rectangulaire.

Indiquez ici les détails qui vous paraissent importants concernant les commandes spécifiques du logiciel de dessin par points que vous utilisez: son nom, les outils disponibles, certaines commandes ou primitives particulières ou intéressantes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

4. L'exécutant "dessin vectoriel"

A. Une solution alternative de numérisation

Vous vous doutez bien que la transformation d'un dessin en pixels entraîne une numérisation très gourmande. Numérisé, un dessin qui occupe une page de format A4 est donc beaucoup plus volumineux qu'une page de texte numérisée du même format. Un autre inconvénient ne vous a pas échappé à l'utilisation: il en est du dessin par points comme de la peinture, tout nouveau trait, toute nouvelle "couche" vient irrémédiablement recouvrir l'ancienne (si l'on excepte la possibilité de plus en plus courante de demander des "retours en arrière" puisqu'on peut programmer l'exécutant pour qu'il mémorise un certain nombre des états antérieurs). Il n'est en tous cas pas possible de modifier les formes, mais plutôt de "retoucher" éventuellement le dessin, pixel par pixel.

Pourriez-vous imaginer une autre manière plus économique de coder (transformer en chiffres) les deux dessins qui précèdent? Décrivez-la du mieux que vous pouvez.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Quelles seraient, d'après vous, les conséquences de cette autre façon de coder sur les primitives attendues des exécutants fonctionnant de la sorte?

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

L'exemple du disque noir est le plus simple. Il suffit de préciser sa couleur (au moyen d'un code, bien sûr), la coordonnée de son centre et son rayon et de fournir à l'exécutant la formule mathématique qui permettra de le dessiner.

Celui de la croix est un peu plus complexe. Il faudra y voir, par exemple, deux rectangles superposés pour lesquels il faudra aussi préciser, toujours par exemple, la coordonnée du point de départ, la longueur et la largeur ainsi que la formule de dessin.

Bien sûr, s'il nous fallait aller jusqu'au bout de cette numérisation, cela resterait un peu complexe. Mais vous percevez que c'est du domaine du possible et que cela convient bien à un exécutant dont la vocation est de calculer. Vous devinez aussi que le gain de place lié à ce type de codage est considérable. Un autre avantage non négligeable: le dessin n'étant plus considéré comme un ensemble de points, il sera désormais un ensemble de formes et de lignes dont les attributs (couleur, épaisseur et style du trait, couleur ou style de remplissage,...) pourront être modifiés "a posteriori". Plus encore, les dimensions de ces formes pourront changer puisqu'elles sont représentées à partir de formules.

Une remarque importante toutefois, si le dessin est mémorisé comme un ensemble de formules, l'exécution de ces formules produira à la sortie (à l'écran comme à l'imprimante) un dessin par points.

B. Les primitives des logiciels de dessin vectoriel

Vous admettez que dans ce contexte, un dessin est un ensemble de formes et de traits élémentaires pouvant donc être sélectionnés séparément ou en groupes, essentiellement en vue d'en modifier les attributs.

Formes et traits

On retrouve généralement, et à peu de choses près, les mêmes formes et les mêmes traits que ceux qui sont offerts par les logiciels de dessin par points. La différence se situe surtout au niveau des modifications ultérieures possibles car ces formes et ces traits ne sont plus codés comme des ensembles de pixels. On parlera donc très volontiers d'objets pour les désigner.

Il n'est plus question de gomme. En effet, chaque objet pouvant être sélectionné, il peut tout aussi bien être supprimé. Vous devinez que les stratégies d'édition seront assez différentes de celles utilisées avec les logiciels de dessin par points.

Il n'est plus question de zoom, dans le sens où celui-ci autorise une modification du dessin, pixel par pixel. Si une commande de zoom existe, c'est uniquement pour faciliter le travail de l'utilisateur et lui donner une vue plus ou moins précise de son dessin.

Sélectionner, couper, copier, coller

Il ne s'agit plus, lors d'une sélection, de définir un ensemble de pixels, mais de choisir un ou plusieurs objets. Le *couper*, *copier*, *coller* reste évidemment d'actualité.

Il est utile de découvrir les techniques permettant la sélection d'un ou plusieurs objets. Pour un seul objet, un simple clic fait souvent l'affaire. Pour en sélectionner d'autres, l'utilisation conjointe d'une touche du clavier est nécessaire car la sélection d'un nouvel objet entraîne la suppression de la sélection du précédent. Si les objets sont proches les uns des autres, des techniques de sélection liées à la définition d'une zone (rectangulaire par exemple) sont quelquefois disponibles. Un objet est sélectionné lorsqu'apparaissent des poignées qui permettent de le saisir.

La sélection des objets ne sert d'ailleurs pas qu'au *couper, copier, coller*. Les objets peuvent être déplacés et leurs dimensions modifiées.

Texte

L'outil texte est aussi présent avec, cette fois, l'opportunité d'une édition (insertion, suppression, modification des attributs) puisqu'il s'agit d'un objet texte.

De nouvelles primitives

Nous avons déjà évoqué la possibilité de sélectionner des objets, puis de les déplacer, d'en changer les dimensions. Selon les circonstances, il est quelquefois possible de leur faire subir d'autres transformations telles: les faire tourner ou de les déformer. Lorsque plusieurs objets sont sélectionnés, on doit pouvoir les grouper et à l'inverse les dissocier. Enfin, et ce n'est pas limitatif, vous pouvez aussi jouer sur les différents plans puisque les objets se superposent les uns aux autres. Vous trouverez une description assez détaillée de ces primitives dans F. SASS, E. VANDEPUT *FP 8.1: Dessin assisté par ordinateur: les actions de base* CeFIS - ICAFOC - FPE 1994

Comment réparer les gaffes?

La gomme n'est plus d'application pour les raisons que nous avons expliquées. En revanche, des options du style *Défaire* ou *Annuler* ont toujours leur raison d'être. Vient s'ajouter à cela la possibilité de supprimer purement et simplement les objets sélectionnés et rien qu'eux.

Le choix du type de logiciel

Selon les travaux à accomplir, le logiciel de dessin par points sera préféré au logiciel de dessin vectoriel ou inversement. Le choix n'est pas irréversible dans la mesure où il existe des logiciels de conversion des dessins d'une catégorie dans l'autre. Toutefois, s'il est assez simple de convertir des dessins vectoriels en points, vous comprendrez que l'inverse est plus complexe et ne fournit pas toujours des résultats très satisfaisants.

Indiquez ici les détails qui vous paraissent importants concernant les commandes spécifiques du logiciel de dessin par points que vous utilisez: son nom, les outils disponibles, certaines commandes ou primitives particulières ou intéressantes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....