REVUE DE STATISTIQUE APPLIQUÉE

MICHEL BETUING
PHILIPPE BILLOT
OTTO MULLER

Les systèmes intégrés de traitement statistique

Revue de statistique appliquée, tome 20, n° 4 (1972), p. 13-30
<http://www.numdam.org/item?id=RSA_1972__20_4_13_0>

LES SYSTÈMES INTÉGRÉS DE TRAITEMENT STATISTIQUE

Michel BETUING, Philippe BILLOT et Otto MULLER *

I - INTRODUCTION

Au cours des vingt dernières années l'un des plus importants facteurs de progrès en matière de traitement statistique a été le développement des possibilités de calcul offertes par des ordinateurs de plus en plus puissants et rapides.

Avec la mise à la disposition des statisticiens de cet outil nouveau sont apparues les tâches de programmation et les problèmes que celles-ci posaient, tant et si bien que de nombreux statisticiens se sont transformés en programmeurs, perdant ainsi de vue l'objet propre de leur science.

Le souci de permettre aux chercheurs en sciences humaines de se dégager de ces contraintes informatiques a été à l'origine d'une évolution qui, sans être terminée, n'en a pas moins permis de faire des progrès considérables dans la simplification des instructions d'utilisation des programmes. Si l'on reprend l'histoire de cette évolution, on constate que le premier besoin qui a été ressenti a été celui de la constitution d'une "bibliothèque de programmes". En effet, auparavant pour chaque problème le chercheur programmait ou faisait programmer le traitement correspondant, ce qui amenait à écrire à plusieurs endroits des programmes remplissant les même fonctions.

Pour éviter cette perte de temps, les instituts de recherche ont été amenés à rassembler les programmes correspondant aux besoins de leurs études, avec le but de disposer d'un ensemble couvrant la variété des besoins. L'avantage de cette solution, la bibliothèque de programmes, est de permettre le choix du meilleur programme statistique pour chaque problème particulier et de profiter rapidement, pourvu que l'information circule bien, des derniers progrès en matière de traitement [1].

---

(1) Le problème de l'information est à l'origine du lancement de projets tels que : "A National Program Library and Central Program Inventory Service for the Social Sciences" par l'Université de Wisconsin ou de l'enquête lancée sous l'égide du Conseil International des Sciences Sociales par l'U.E.R. de Mathématiques, Logique Formelle et Informatique de l'Université René Descartes (Paris).

* Atelier Parisien d'Urbanisme (APUR)

Les auteurs tiennent à remercier Monsieur de Hemptinne directeur de la division de la Politique Scientifique (Unesco), et F.M. Andrews (Université de Michigan) sans qui ce travail n'aurait pas été possible.

Au regard de cet avantage, les inconvénients restent nombreux :

- les programmes provenant de différentes sources, il n'y a aucune unité de conception dans leur écriture ;

- ils sont souvent écrits dans des langages différents ;

- ces deux caractéristiques entraînent de sérieuses difficultés dans la résolution des problèmes qui se présentent lors de l'utilisation des programmes ;

- de plus, il faut en général modifier chaque fois les programmes lorsqu'on change de données.

Le souci de réduire les difficultés précédentes et en particulier d'éliminer la partie de programmation qui restait toujours à faire, a amené à concevoir des ensembles cohérents de programmes dont la particularité essentielle sur le plan de l'utilisation est de fonctionner à l'aide d'instructions codées : il existe un modèle d'instructions valable pour tous les programmes de l'ensemble et dont la syntaxe est facilement assimilable par les non-spécialistes.

On peut appeler de tels ensembles des "paquets de programmes standardisés". L'exemple le plus connu en est l'ensemble B.M.D. (Biomedical Computer Programs) de l'Université de Californie qui a été conçu, comme son nom l'indique, pour les besoins de la recherche médicale (1).

L'inconvénient de ces paquets statistiques est qu'il leur manque en général toute la partie des traitements qui concernent l'organisation des données que l'on peut avoir à faire à partir des résultats bruts d'une enquête. En effet, les paquets sont conçus pour traiter des fichiers bien définis (2). Or, dès que l'on doit traiter une enquête on se heurte à un certain nombre de problèmes matériels tels que :

- recherche des cartes manquantes pour une observation ;

- tri et sélection des observations ;

- vérification des codes ;

- recodage ;

- traitement des données manquantes ;

- etc...

Ensuite viennent les traitements statistiques proprement dits qui nécessitent très souvent des manipulations de fichiers, que l'on veuille par exemple

---

(1) B.M.D. - Biomedical Computer Programs - W.J. Dixon, Editor. University of California Press (1ére version parue en 1960).

(2) De plus, il est encore nécessaire de se préoccuper des questions de format des données à chaque utilisation d'un programme, ce qui est assez contraignant lorsqu'on veut enchaîner des traitements sur des données de départ. Il faut en effet pouvoir préciser quel sera le format des résultats du 1er traitement pour leur appliquer le 2e traitement, etc...

ne prendre en compte dans un traitement que certaines variables ou certaines observations, ou créer de nouvelles variables à partir des premiers traitements effectués.

Si la résolution de tous ces problèmes ne présente pas de difficultés insurmontables, il n'en demeure pas moins qu'elle demande un certain temps et est elle-même source d'erreurs.

La phase suivante d'évolution consiste alors à faire prendre en compte de façon automatique par l'ordinateur toutes ces manipulations de fichiers et de programmes et à ne laisser à l'utilisateur que la préoccupation de la définition des opérations à effectuer. C'est pourquoi, on définira les "systèmes intégrés de traitements statistiques" comme des ensembles de programmes permettant, au moyen de procédures standardisées, de définir les opérations qu'on désire faire exécuter dans un langage se rapprochant le plus possible de celui que le chercheur en sciences humaines et sociales est habitué à employer.

Cette facilité de mise en oeuvre est obtenue :

- en rendant compatibles tous les programmes qui peuvent être nécessaires pour le traitement statistique d'une enquête ;

- en faisant prendre en compte la gestion du fichier que l'on traite par un sous-programme que l'on utilise préalablement à tout traitement statistique.

Ainsi les programmes statistiques peuvent être indépendants entre eux, leur lien sera ce programme de gestion permettant toujours de revenir au fichier.

Ayant précisé ce qu'était un système intégré, nous allons maintenant comparer les principaux systèmes disponibles actuellement sur la base des critères suivants :

A - Les fonctions remplies, c'est-à-dire l'ensemble des traitements que l'on peut faire sur un fichier de départ provenant d'une enquête.

B - Les caractéristiques de diffusion, c'est-à-dire l'adaptabilité à d'autres types d'ordinateurs que celui pour lequel le système intégré a été mis au point à l'origine, les conditions de mise à jour et d'adjonction des programmes et enfin les coûts.

C - Les caractéristiques d'utilisation des programmes.

II - COMPARAISON DES SYSTEMES INTEGRES LES PLUS REPANDUS : DATA-TEXT, OSIRIS et SPSS

Il existe un grand nombre de systèmes intégrés, plus ou moins complets du point de vue des fonctions remplies (1), tant dans les universités que chez certaines sociétés de service (2).

Il est impossible ici de les comparer dans leur ensemble. L'un des critères de comparaison retenus étant la diffusion (3) de ces systèmes, on n'étudiera que ceux qui sont les plus répandus auxquels correspondent d'ailleurs les manuels d'utilisation les plus au point (4). Les systèmes pris en considération sur la base des résultats des enquêtes de E.D. Meyers et H.F. Cline sont les suivants (5) :

---

(1) Nous ne nous intéressons ici qu'aux systèmes de traitements statistiques.

(2) En ce qui concerne les universités américaines, il existe un premier recensement des systèmes de programmes dans le document suivant : Survey of Social Science Computing Systems, Edmund D. Meyers Jr, Project Impress, Dartmouth College, Hanover, N.H. (1969).
Pour l'Europe une enquête, déjà citée, est actuellement menée par l'U.E.R. de Mathématiques de l'Université René Descartes (Paris), sous l'égide du Conseil International des Sciences Sociales. Cette enquête porte sur les programmes utilisés en Sciences Humaines, ce qui déborde du cadre des programmes de traitement statistique.

(3) Rappelons qu'il existe deux ensembles de programmes très répandus : SSP : Scientific Subroutine Package, fourni par IBM, B.M.D. Biomedical Computer Programs (Université de Californie) Mais d'une part, ce ne sont pas des systèmes intégrés au sens où nous les avons définis, d'autre part, ils n'ont pas été conçus pour les sciences sociales.

(4) Voir à ce sujet : 1. L'étude d'Edmund D. Meyers déjà citée. 2. Study of Computer Uses in Social Science Research. Hugh F. Cline, Russell Sage Foundation.New-York 1970.

(5) Comme le fait remarquer Cline, de nombreux systèmes ne sont que des variantes de ceux que nous allons considérer. Il existe au Royaume Uni un système intégré, ASCOP (ASCOP - A Statistical Programming Language - mis au point par le National Computing Centre (N.C.C.), Quay House, Quay Street, Manchester 2. Ce système est actuellement opérationnel sur la série 1900 des calculateurs I.C.L. Il est donc de ce fait peu diffusé en dehors de la Grande Bretagne. Son coût d'achat est relativement élevé (de l'ordre de 55 000 F.F.), d'autant plus qu'il lui manque la partie gestion de fichiers. Le système SALY de l'Université d'Essex comprend uniquement des programmes d'analyse statistique, dont une partie proviennent de OSIRIS, système qui figure parmi ceux retenus pour la comparaison. (Saly User's Manual, Computing Center, University of Essex, 1970).

En France il existe un certain nombre de systèmes intégrés tels que NLT (SOGREAH - Institut International d'Informatique - Grenoble), DAPHNE (SEMA Paris), G.T.S. (mis au point à l'INSEE et diffusé par le CAP, Centre d'Analyse et de Programmation, Paris), PRALINE (I.B.M.), etc... Ces systèmes sont principalement axés sur le dépouillement d'enquêtes et ne comprennent pas encore de traitements statistiques élaborés. La plupart d'entre eux ne sont pas d'ailleurs vendus mais utilisables en service bureau.

A l'Atelier Parisien d'Urbanisme (APUR) il a également été mis au point un système intégré (PROFIL), mais celui-ci ne prend en compte que des fichiers à structure bien définie découlant des besoins propres aux études urbaines et par ailleurs ne contient que des programmes d'analyse statistique.

- S.P.S.S. Statistical Package for the Social Sciences. NORC (National Opinion Research Center), Université de Chicago.

- DATA-TEXT : Department of Social Relations - Université de Harvard ;

- OSIRIS : Organized Set of Integrated Routines for Investigations with Statistics ;

OSIRIS existe en deux versions :

. OSIRIS I, produit par : the Institute of Social Research (I.S.R.), Ann Arbor, Michigan.

. OSIRIS II, produit par : the Inter-University Consortium for Political Research (ICPR), Ann Arbor, Michigan.

A. FONCTIONS REMPLIES

On en distinguera trois types, bien que la délimitation entre les deux derniers soit assez difficile : gestion de fichiers, statistique descriptive, analyse statistique. Tout traitement d'enquêtes désirant aller plus loin que la simple description et interprétation monographique se doit d'avoir recours successivement, et avec d'éventuels retours en arrière, à ces trois types de fonctions. C'est pourquoi nous n'avons retenu que des systèmes les contenant (1).

1. Gestion de fichiers

a) Préparation des données

Cette fonction comporte normalement les étapes suivantes :

- vérification de classement et éventuellement mise sur bande si le fichier est de taille importante.

Cette vérification a pour but de :

. regrouper les cartes perforées correspondant à une même observation,

. les ordonner suivant le format du fichier,

. repérer les cartes manquantes ou les cartes dédoublées. Le programme peut soit lister ces cartes, soit créer des cartes fictives pour remplacer celles qui manquent, et éliminer les cartes en double.

- Construction des fichiers standards tels qu'ils seront traités par le système. En effet, un système intégré nécessite que le fichier se présente sous une forme bien définie. Pour OSIRIS, par exemple, il faut donner un nom et décrire les variables à l'aide d'un "dictionnaire". Pour SPSS et DATA-TEXT, il faut préciser le format FORTRAN.

- Vérification des codes. Correction de fichiers. Le programme vérifie que les codes correspondent bien à des valeurs possibles et liste les erreurs. Il y a alors plusieurs options :

---

(1) voir note (1) page 16.

. correction automatique,
. élimination des observations correspondant à des codes erronés
. correction manuelle.

b) Transformation des données. Opérations portant sur les variables

Cette fonction permet de créer de nouveaux fichiers à partir d'un ou plusieurs fichiers au moyen des opérations suivantes :

- création de nouvelles variables ou d'indices par transformation, au moyen des fonctions usuelles, sur des variables de départ,

- recodage,

- fusion de fichiers,

- suppression de variables.

c) Sélection de sous-fichiers et agrégations. Opérations portant sur les observations

Cette fonction comprend :

- des programmes de tri permettant de sélectionner des sous-fichiers ;

- des programmes d'agrégation permettant, à partir de plusieurs observations, de créer une observation d'un autre type, en général à un niveau plus élevé.

Sur l'ensemble de ces trois fonctions, le système SPSS est moins intéressant que les trois autres qu'il est plus difficile de différencier, si ce n'est à accorder un léger avantage à DATA-TEXT en ce qui concerne la fonction a), à OSIRIS I et II pour la fonction b) et à OSIRIS I pour la fonction c) (1).

2. Statistique descriptive

Les fonctions qu'on inclut dans cette catégorie (2) sont les suivantes:

a) Tris à plat, c'est-dire étude de la distribution d'une variable sur l'ensemble des observations retenues et calcul des caractéristiques habituelles : étendue, moyenne, médiane, écart-type, etc...

---

(1) Comme il ressort de la lecture des documents suivants :
- SPSS : Statistical Package for the Social Sciences. Nie, Bent, Hull - 1970 Mac Graw Hill auquel on fera référence en tant que "manuel SPSS" ;
- Description of the OSIRIS/40 Programs. ISR University of Michigan - January 1972.
- A brief synopsis of programs in OSIRIS II, Level 2, University of Michigan - march 1971.
- Data analysis systems. A user's point of view, Klaus Allerbeck. Communication présentée au "Workshop on computer programming systems for the social sciences", Paris, avril 1971

(2) Bien que nous soyions conscients de l'arbitraire de cette classification. Disons que c'est la partie de la statistique à laquelle se limitent actuellement la plupart des langages de dépouillement d'enquêtes.

b) Tris croisés : édition des tableaux croisant deux ou plusieurs variables et étude de ces tableaux (calculs de fréquences, fréquences conditionnelles, marginales, mesures non paramétriques de relations)

c) Représentation graphique. Cette partie comprend le tracé d'histogrammes, de graphes de fonctions de répartition cumulées, de nuages de points..

En ce qui concerne les fonctions a et b, tous les systèmes font à peu près la même chose, OSIRIS I et II étant cependant moins performants, en ce sens qu'ils ne peuvent pas croiser plus de quatre variables alors que les autres peuvent aller jusqu'à 7 ou 8. Etant donné qu'il existe des méthodes d'analyse multidimensionnelle, ce défaut n'est pas rédhibitoire.

Quant à la fonction c aucun des systèmes étudiés ne semble la remplir de façon satisfaisante.

3. Analyse statistique

Du point de vue des principales fonctions remplies dans ce domaine, les systèmes étudiés se présentent de la façon suivante :

	DATA TEXT	SPSS	OSIRIS I	OSIRIS II
Corrélations :				
coef. Bravais - Pearson	x	x	x	x
c. multiples	x	x	x	x
c. partielles		x	x	x
c. des rangs		x	x	x
Régression multiple	x	x	x	x
Analyse de variance	x		x	x
Analyse en composantes principales	x	x	x	x
Rotation d'axes	x	x	x	x
Echelles de Guttman		x		x
Segmentation			x(1)	x(1)
Typologie (clustering)			x	x
Statistiques non-paramétriques autres que la corrélation			x	
Tests de Student, Fisher (2)	x		x	

(1) Il s'agit des programmes AID et MCA.

A.I.D : Automatic Interaction Detection
M.C.A.: Multiple Classification Analysis

Pour A.I.D. voir : The detection of interaction effects. J.A. Sonquist. J.N. Morgan

Pour M.C.A. Multiple. Classification Analysis. F.M. Andrews J.N. Morgan J. A. Sonquist.

Publications de : I.S.R. University of Michigan. Ann Arbor.

(2) Il va de soi qu'il s'agit de programmes qui fonctionnent séparément ce qui ne préjuge en rien du fait que les programmes d'analyse de variance ou autres peuvent contenir des tests dans leur algorithme.

B. CARACTERISTIQUES DE DIFFUSION

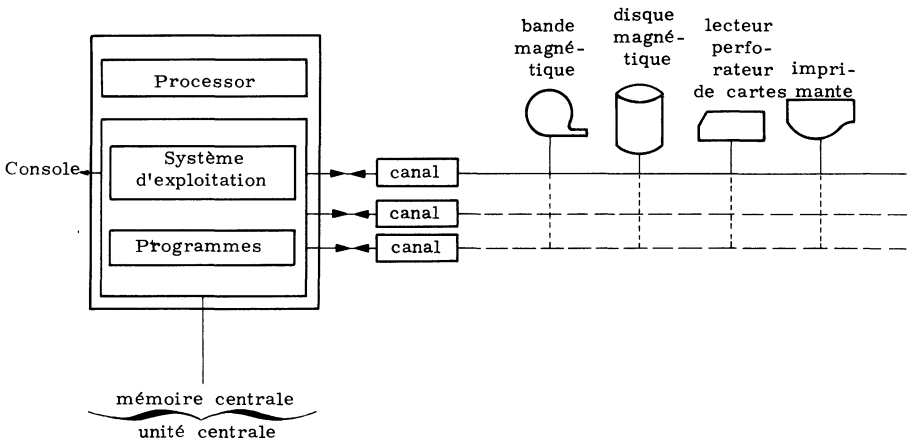
Les problèmes posés par la diffusion des systèmes intégrés se situent à trois niveaux :

- implantation,
- mise à jour,
- adjonction de programmes.

Pour comprendre ces problèmes, il est nécessaire de rappeler brièvement comment se présente un ordinateur et ce que sont les caractéristiques informatiques d'un système intégré.

1. Configuration générale d'un ordinateur

a - Organisation physique et fonctions générales



On distingue :

- l'unité centrale, composée de la "mémoire centrale" et du "Processor". Le processor est l'organe de traitement et de calcul chargé d'exécuter les instructions communiquées par le programme ; instructions et informations à traiter doivent être en mémoire centrale.

- les unités périphériques qui sont de deux types :

. unités séquentielles : les enregistrements ne peuvent être lus que dans l'ordre où ils ont été écrits (lecteur perforateur de cartes, unités à bandes magnétiques, imprimantes..),

. unités à accès direct : la relecture des enregistrements peut se faire dans un ordre différent de leur écriture (disques et tambours mangétiques...)

Ces unités périphériques sont reliées à la mémoire centrale par l'intermédiaire de canaux. Le canal a pour rôle, sur ordre du Processor, d'effectuer un échange entre l'unité périphérique qu'on lui a désignée et la mémoire centrale. Celle-ci occupe une place fondamentale puisque c' est par elle que transitent automatiquement instructions et informations. On y trouve à la fois le "Système d'Exploitation" (qui travaille en mode "maître") et les programmes utilisateurs (qui travaillent en mode "esclave", c'est-à-dire que pour toute opération autre que les opérations algébriques élémentaires ils doivent déposer une requête auprès du système d'exploitation (1)). Dans le cas du système intégré, il n'est pas nécessaire que l'ensemble des programmes utilisateurs figurent en mémoire centrale. Par une commande (exemple : carte contrôle ou appel de sous-programme), on avertit le système d'exploitation qu'on désire exécuter un certain programme. Dès que cela lui est possible, le S.E. charge en mémoire le programme cité et lui donne le contrôle, jusqu'à ce que celui-ci soit terminé ou réclame un autre programme, etc... Tout programme est ainsi lié au système d'exploitation pour lequel il a été conçu.

b - Communication homme-machine

Les ordinateurs sont conçus actuellement suivant la logique binaire ou booléenne (le courant passe ou ne passe pas). Tout programme pour être exécutable par une machine donnée doit, au stade ultime, être codé instruction par instruction dans le système binaire. Programmer directement en binaire est un travail considérable aux risques d'erreurs très importants. Le stade le plus élémentaire de programmation pratiqué est en fait la programmation en "langage assembleur", qui est une écriture symbolique des instructions binaires usuelles. Un programme faisant partie du système d'exploitation se charge de traduire ces instructions du langage assembleur en instructions binaires. Toutefois ce langage, qui permet d'exploiter à fond toutes les possibilités de la machine, n'est pas encore assez généralisé pour pouvoir faire facilement des opérations élaborées. Le calcul d'une expression arithmétique, par exemple, nécessite toute une séquence d'instructions écrites en assembleur.

Pour se dégager davantage des contraintes technologiques de la machine et en même temps se rapprocher des besoins de l'utilisateur, on a mis au point une nouvelle classe de langages de programmation ; les langages évolués tels que FORTRAN, ALGOL, PL1, etc... La composition des instructions de ces langages est soumise à certaines règles de grammaire et de syntaxe. Avant de pouvoir exécuter les fonctions décrites dans un tel langage il faudra procéder à deux étapes préliminaires :

- compiler : traduire les instructions du langage évolué en assembleur.

- assembler le jeu d'instructions résultant et charger le code binaire généré en mémoire.

---

(1) celui-ci est aussi sollicité dans le cas d'erreur d'exécution d'une instruction simple.
Les I.B.M. 360 utilisent deux systèmes :
D.O.S. : Disk Operating System
O.S. : Operating System ;

De tels langages permettent alors de communiquer avec n'importe quel ordinateur pourvu que celui-ci soit muni d'un compilateur adéquat. Pour certains langages (par exemple FORTRAN) tous les fabricants ont créé un tel compilateur, pour d'autres il n'existe de compilateur que sur des ordinateurs particuliers (par exemple, le compilateur PL1 n'existe que pour la gamme d'ordinateurs IBM 360 et au dessus). Notons que plus le langage est évolué, plus il permet de faire de façon simple des opérations compliquées, mais plus il nécessite de taille mémoire et moins il est performant. Les langages couramment utilisés (FORTRAN, COBOL) réalisent un compromis entre leur degré d'évolution et leur coût en exécution (1) par le fait qu'ils sont spécialisés (2). Une bonne programmation doit utiliser les divers niveaux de langages pour assurer toutes les fonctions et minimiser les coûts.

2. Caractéristiques informatiques du système intégré

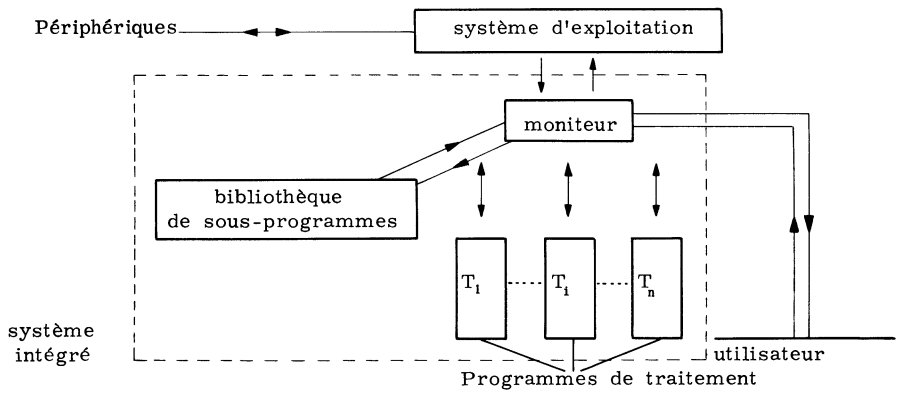
Elles sont de deux ordres :

- leur organisation logique,
- les langages qu'ils utilisent

a - Organisation logique

On distingue :

- les systèmes à "moniteur"



Dans un tel système intégré, le Moniteur est le programme chargé de recevoir les commandes de l'utilisateur, de solliciter le programme de trai-

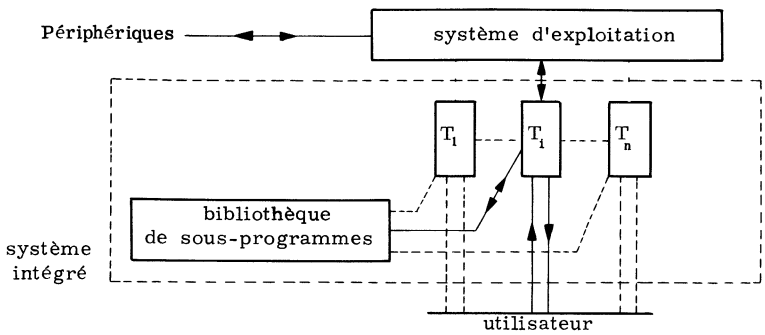
---

(1) Ainsi l'ALGOL qui est un langage très évolué n'est que peu commercialisé en raison de la complexité et du coût de sa compilation.

(2) Par exemple FORTRAN pour le calcul scientifique, COBOL pour la gestion et les manipulations de fichiers.

tement adapté, de contrôler les échanges externes via le système d'exploitation, et enfin de répercuter les résultats vers l'utilisateur. Celui-ci n'a donc affaire qu'au programme moniteur.

- les systèmes multi-programmes



L'utilisateur appelle lui-même le programme qu'il désire. Ce programme s'exécute indépendamment de ceux qui ont pu le précéder ou de ceux qui vont le suivre.

- Comparaison de ces deux types d'organisation

Un système avec moniteur présente les avantages suivants :

. Il offre une plus grande sécurité d'utilisation. Dans une suite de traitement il y a un contrôle permanent de la bonne exécution et du juste enchaînement des tâches.

. L'optimisation des traitements est assurée, dans la mesure où le moniteur peut éviter des calculs redondants. Par exemple si l'on a déjà calculé la matrice de corrélations sur des données et si l'on désire faire passer une analyse en composantes principales sur ces mêmes données un bon moniteur réutilisera la matrice de corrélations déjà trouvée.

. Par contre, une erreur de programmation dans le système intégré sera plus difficile à localiser dans un système avec moniteur que dans un système multi-programmes.

. De plus, un système avec moniteur nécessitera une taille moyenne[1] de mémoire centrale plus importante. En effet, un moniteur doit pouvoir a priori assurer l'initialisation et le contrôle de n'importe quelle tâche et il doit par ailleurs rester en mémoire centrale pendant toute la durée du traitement.

b - Les langages utilisés

Comme on l'a déjà vu, un système intégré remplit trois grands types de fonctions :

---

(1) ou plutôt des tailles minimale et maximale.

- accès aux fichiers,

- maintenance de ces fichiers (tri, fusion de fichiers, extraction de sous-fichiers, ...),

- calculs.

L'optimisation d'un système conduit à programmer ces différentes fonctions dans des langages divers : FORTRAN pour les calculs, PL1 COBOL ou ASSEMBLEUR pour les accès aux fichiers, ASSEMBLEUR ou sous-programmes du "système d'exploitation" pour la maintenance.

c - Ce que réclame un système intégré

- Un système intégré est étroitement lié au système d'exploitation pour lequel il a été conçu et donc à un type d'ordinateur. La présence de parties écrites en Assembleur renforce cette contrainte. Quant aux différents langages évolués utilisés, ils requièrent chacun l'existence du compilateur correspondant.

- Préalablement à toute exécution, il faut pouvoir disposer d'une place en mémoire centrale au moins égale à celle que réclame le plus encombrant des programmes du système dans la mesure où on veut pouvoir utiliser toutes les fonctions.

- Au niveau même de l'exécution, il est nécessaire de fournir au système intégré un certain nombre de mémoires périphériques (disques ou tambours pour fichiers à accès direct, bandes magnétiques ou disques pour fichiers séquentiels, lecteurs de cartes et imprimantes pour rentrer les paramètres et éditer les résultats...).

3. Implantation

Implanter un système ou un programme revient à le rendre exécutable sur une machine donnée. Le problème est assez différent suivant que la transplantation porte sur des configurations compatibles ou non.

a - La configuration receveuse est compatible avec la configuration origine

C'est l'ensemble de la machine et du système d'exploitation qui fait que deux configurations sont compatibles ou non. Ainsi tous les ordinateurs de la série des IBM 360 sous la même version (release) d'Operating System sont compatibles entre eux. Il en va de même pour tous les 360 sous la même version de DOS. Par contre, entre une configuration de la première série et une de la seconde, il n'y a pas de compatibilité. Dans le cas de compatibilité les opérations auxquelles il faut procéder sont les suivantes :

- chargement du jeu binaire créé sur la configuration origine dans la bibliothèque courante ou création d'une nouvelle bibliothèque,

- préparation et initialisation d'un certain nombre d'espaces de travail,

- Eventuellement extension de certaines facilités offertes par le système d'exploitation (exemple : procédures cataloguées).

b - La configuration receveuse est différente de la configuration origine

Suivant le degré de différence, on aura au minimum à recompiler l'ensemble de tous les programmes du système intégré, ou même à réécrire l'ensemble de tous les programmes et sous-programmes. L'utilisation extensive de langages évolués dans le système a pour conséquence de limiter les risques de réécriture (mais comme on l'a déjà vu cela rend le système moins performant). Ainsi le passage d'un système écrit pour un IBM 360 OS à un CDC 6600 va obliger à réécrire toutes les parties programmées en Assembleur en Compass, à refaire en Fortran ou Algol les modules écrits en PL1 et à recompiler le tout. Dans le cas particulier d'un système avec moniteur, il est possible qu'on ait à intervenir également dans la logique de ce moniteur. On peut donc très vite arriver à des tâches considérables et coûteuses et un système est d'autant plus intéressant pour un acquéreur éventuel qu'il en existe déjà une version tournant sur une configuration compatible avec la sienne propre.

4. Mise à jour

On appelle ainsi toute modification due à l'organisme créateur du système. Ces modifications peuvent consister en :

- l'adjonction d'un programme nouveau,
- la suppression d'un programme ancien,
- la modification d'une séquence d'instructions dans un programme.

A ce propos, il faut remarquer que :

- les modifications sont toujours faites par rapport à la version origine et non par rapport à des versions modifiées par les utilisateurs.

- elles peuvent prendre deux formes :

. fourniture de la nouvelle version complète du système incluant les mises à jour ;

. envoi des mises à jour détaillées à faire par l'utilisateur lui-même.

Au niveau des problèmes posés, on retrouve la même distinction qu'au paragraphe précédent.

5. Adjonction de programmes

Il s'agit ici de programmes ajoutés par l'utilisateur pour répondre à des besoins qui lui sont particuliers. (1)

Les problèmes sont différents suivant qu'on a affaire à un système avec ou sans moniteur. Dans le premier cas, il faudra en effet modifier le moniteur pour qu'il prenne en compte le nouveau programme et qu'il en assure

---

(1) Par exemple on peut désirer adjoindre l'analyse des correspondances du professeur Benzecri à de tels systèmes.

le contrôle. Les problèmes de programmation sont alors compliqués et les risques d'erreurs importants, et cela suppose une connaissance approfondie du système sur le plan informatique. Les problèmes ne sont pas évidents, mais plus simples lorsqu'il s'agit d'un système multi-programmes, puisqu'alors la connaissance des sous-programmes (fonctions et procédures d'appel) et de la structure des fichiers suffit pour introduire un nouveau programme.

6. Comparaison des systèmes (1)

Pour faire cette comparaison, on va d'abord rappeler les caractéristiques des systèmes étudiés par rapport aux différents points évoqués et on y ajoutera les caractéristiques de coût et de diffusion actuelle. Il faut noter au sujet du système DATA-TEXT que celui-ci se présente sous deux versions. La première est écrite pour les ordinateurs de deuxième génération, la seconde (dont la mise au point est en cours d'achèvement) est destinée aux ordinateurs de troisième génération. Lorsqu'on parle de DATA-TEXT, sans autre précision, il s'agit de la version 3e génération.

On voit donc que les coûts des systèmes étudiés sont du même ordre. Le système dont les versions actuelles sont les plus nombreuses, c'est-à-dire qui pourra être implanté directement le plus facilement, est SPSS. Les mises à jour de ce système sont également les plus fiables et les plus faciles à faire puisqu'on fournit une nouvelle bande complète. Elles seront par contre en général plus coûteuses parce qu'il faut refaire toutes les opérations d'implantation. Il faut noter d'autre part que OSIRIS II va combler une partie de son retard très prochainement lorsque les versions en cours seront terminées et d'ailleurs OSIRIS II est malgré tout le système le plus répandu en Europe Les systèmes les moins exigeants en taille mémoire sont les deux OSIRIS. Ceci est lié, comme on l'a vu, au fait que ce sont des systèmes multi-programmes tandis que les deux autres utilisent un moniteur (2).

C. CARACTERISTIQUES D'UTILISATION

L'utilisation d'un système intégré pour les besoins du traitement d'une enquête comporte deux phases : la description du fichier au système, les traitements proprement dits.

---

(1) Sur la base des documents suivants :

- Survey of Social Science. Computing Systems (voir note 2 page 16)
- Data Analysis Systems. Klaus R. Allerbeck (voir note 1 page 18)
- Study of Computer Uses in Social Science Research Hugh F. Cline (voir note 4 page 16)
- OSIRIS I. Description of the OSIRIS/40 programs. Release 6. ISR Michigan
- OSIRIS II. A brief synopsis of programs in OSIRIS II, level 2 ICPR
- SPSS. Manuel d'utilisation Nie-Bent - Hull

(2) Rappelons que la dernière version d'OSIRIS I (janvier 1972) est maintenant pourvue d'un petit programme moniteur, mais cela ne change rien à l'allure générale du système.

Critères de Comparaison	Systèmes / Nature	DATA-TEXT avec moniteur	SPSS avec moniteur	OSIRIS I multi-programmes (1)	OSIRIS II multi-programmes
Langages utilisés par la version origine		Assembleur Fortran	Assembleur Fortran	Assembleur Fortran	Assembleur Fortran PL 1
Taille mémoire centrale Min K. octets Max K. octets		 200 250	 160 230	 100 128	 100 200
Version origine Autres versions opérationnelles		IBM 360.OS CDC 6600	IBM 360 OS CDC 6600 Univac 1108 RCA Spectra 70/45/46/60 PDP 10 Burroughs 6500	IBM 360 OS	IBM 360 OS
Versions en cours de mise au point					CDC 6600 SIEMENS RCA
Type de mise à jour		Système en fin de réécriture pour 3ème génération	Nouvelle bande fournie par le concepteur	Nouvelle bande fournie par le concepteur	Adjonctions faites par l'utilisateur
Coûts (2) Achats Maintenance		 $ 750	 $ 400 $ 200	 $ 500 $ 200	 $ 950 $ 475
Diffusion universitaire en Europe (listes non exhaustives)		2ème génération Allemagne : Cologne Grande-Bretagne : Londres	Allemagne : Cologne Ecosse : Edinburgh France : Paris/Orsay Pays-Bas : Leyden	Paris/Orsay (CIRCE) (3) Paris/APUR (4)	Allemagne : Cologne Mannheim Belgique : Louvain France : Paris/Orsay Norvège : Bergen Pays-Bas : Amsterdam Suède : Göteborg Turquie : Istambul

(1) La version 6 d'Osiris I est maintenant pourvue d'un petit moniteur dont le rôle semble se borner à appeler les différents programmes consécutivement. Le moniteur décharge l'utilisateur d'une partie des préoccupations liées au système d'exploitation : une seule procédure cataloguée, réduction du nombre de cartes contrôlées à écrire.

(2) Les coûts indiqués sont ceux pratiqués pour les universités. La distribution d'OSIRIS II peut s'effectuer d'une matière particulière : en tant qu'universitaire, on peut adhérer à une association d'universités (ICPR : Inter-University Consortium for Political Research) qui gère le centre de calcul se trouvant à Ann Arbor Michigan, et où s'élabore OSIRIS II. Pour les membres de ce consortium, le prix d'achat d'OSIRIS II est de $ 300 et le prix de maintenance de $ 150 U.S. Quant aux sociétés commerciales, elles peuvent se procurer OSIRIS II pour $ 1 900 U.S. et en assurer la maintenance pour $ 950.

(3) CIRCE = Centre Inter Disciplinaire Régional de Calcul Electronique (CNRS)

(4) non universitaire.

1. Description

a) Pour qu'un système intégré accepte de traiter un fichier, il faut non seulement que celui-ci se présente sous une forme donnée, mais encore qu'il respecte un certain nombre de contraintes. De ce point de vue, les systèmes possèdent les caractéristiques suivantes :

	DATA-TEXT	SPSS	OSIRIS I	OSIRIS II
Aspect du fichier	Séquentiel cartes	Séquentiel cartes	Sequentiel cartes	Séquentiel cartes
Nb maximum de cartes par observation	illimité	limité par le nb de variables	50	50
Nb de variables maximum	illimité	500	1 600	1 600
Nb d'observations ;	illimité	illimité	illimité	illimité

DATA-TEXT est le système le plus intéressant pour cet ensemble de critères, les possibilités des trois autres étant cependant largement suffisantes dans la plupart des cas.

b) La présentation des données en elle-même varie suivant les systèmes étudiés.

Dans le cas des deux OSIRIS, la présentation se fait de façon préalable à tout traitement. Un programme spécial crée un fichier standard en deux parties :

- un dictionnaire qui contient la liste des variables avec leurs caractéristiques (comme par exemple : numéro de la variable, nom, place occupée dans le fichier cartes de départ, nature, valeur maximum, codes erreurs, etc...),

- le fichier des données.

Pour SPSS et DATA-TEXT, la présentation se fait à l'occasion du premier traitement. On y décrit le fichier cartes de façon habituelle (par exemple sur I.B.M. des instructions de format Fortran), mais le système peut, sur ordre, enregistrer cette description qui ne sera plus à faire par la suite.

Le premier type de description présente sur le second l'avantage de permettre le contrôle des données avant tout traitement statistique, par contre dans ce cas, la durée totale d'utilisation de l'ordinateur est plus longue et par conséquent le coût plus élevé. Ceci est surtout sensible lorsqu'on a beaucoup de fichiers à traiter et peu de traitements à effectuer sur chacun d'eux.

2. Traitements

2.1. Les instructions nécessaires pour assurer ces traitements sont de deux sortes :

a) Celles qui sont données au système d'exploitation de l'ordinateur par le moyen des cartes contrôle dont l'aspect est fixé par le système d'exploitation.

On y précise :

- le nom du programme que l'on veut exécuter ;

- la classe de périphériques où l'on veut situer les mémoires externes ;

- un certain nombre de paramètres renseignant sur la taille, la forme et le statut des fichiers que l'on utilisera sur ces mémoires périphériques (1).

Que les systèmes soient multi-programmes ou à moniteur le nombre de cartes et la quantité de renseignements y figurant en ce qui concerne les périphériques sont du même ordre. Par contre un système avec moniteur ne demande qu'une carte d'exécution (portant le nom d'un unique programme) tandis qu'un système multi-programmes réclame autant de cartes exécution que de programmes désirés. L'un comme l'autre réclament une bonne connaissance du langage du Job Control, mais un système avec moniteur (SPSS, DATA-TEXT) est quand même légèrement plus facile à utiliser (2).

b) Celles qui sont données au système intégré pour remplir les différentes fonctions voulues à l'intérieur d'un même traitement.

Ces instructions doivent être :

- simples : il y a en gros deux types principaux de formulation. Le premier consiste à employer des cartes paramètres (où l'on représente les opérations à effectuer par des codes chiffrés : souvent le code binaire pour indiquer l'absence ou la présence de l'opération à accomplir). Dans le second type, le langage est voisin de celui du chercheur (3).

- homogènes. L'emploi d'instructions suppose une syntaxe. Si celle-ci n'est pas uniforme dans tout le système intégré, c'est-à-dire si des mêmes fonctions peuvent être commandées par des instructions différentes lors de traitements différents, l'apprentissage sera plus malaisé et les risques d'erreurs seront plus grands.

---

(1) Temporaire ou définitif - en création, en lecture ou en mise à jour.

(2) Il faut remarquer que les facilités apportées par les procédures cataloguées peuvent être aussi bénéfiques à un type de système qu'à un autre.

(3) Par exemple pour DATA-TEXT ; COMPUTE CORRELATIONS (XX BY YY), où XX et YY sont des noms ou numéros de variables.

Les deux OSIRIS demandent des cartes paramètres pour appeler les fonctions, tandis que SPSS et DATA-TEXT utilisent un langage à grammaire voisin de celui du chercheur.

De plus, les cartes paramètres d'OSIRIS diffèrent parfois dans leur composition d'un programme à un autre.

2.2. Les messages d'erreur

Une erreur commise dans les instructions donne lieu en général à un diagnostic d'erreur. Plus le message est précis, plus l'apprentissage est aisé. De ce point de vue les systèmes étudiés sont de qualité moyenne et équivalents entre eux. A propos d'OSIRIS on peut noter qu'il détecte seulement les erreurs se trouvant dans le premier programme d'une chaîne de traitements car l'exécution s'arrête aussitôt après.

Les deux systèmes OSIRIS apparaissent donc comme les plus complexes à l'utilisation, mais pour des usages répétés, l'utilisateur acquiert néanmoins assez rapidement la maîtrise de leur syntaxe.

CONCLUSION

Il ne s'agit pas ici d'établir un ordre de préférences sur les systèmes étudiés, d'autant plus que la sélection de ceux-ci pour la comparaison relève déjà d'un certain arbitraire et provient principalement du fait que ce sont les plus répandus (1). Le choix d'un système dépend essentiellement du type de problèmes que l'on a à traiter et du matériel dont on dispose.

En effet, il n'est pas intéressant pour un utilisateur normal de réécrire un système pour l'adapter à son ordinateur car c'est un travail très lourd et peu rentable. Il faut donc au maximum profiter des versions existantes. Lorsqu'on a la disposition d'un matériel adéquat, il est préférable, étant donné leur faible coût, de posséder plusieurs systèmes intégrés : les deux OSIRIS sont les plus intéressants sur le plan du dépouillement d'enquêtes (2) ou des méthodes de segmentation et de façon générale sont les plus complets en ce qui concerne l'ensemble des traitements statistiques (3). Par ailleurs, ce sont les plus aptes à subir des modifications de la part des utilisateurs ; lorsqu'on a affaire à des fichiers sûrs et sous forme de tableau rectangulaire, il peut sembler préférable d'employer SPSS dont la manipulation est plus aisée. Quant à DATA-TEXT, dont la version troisième génération est maintenant opérationnelle, il réclame une grande capacité de mémoire centrale, mais il est également d'utilisation assez simple et permet un grand nombre de traitements.

---

(1) H.F. Cline (Study of Computer Uses, déjà cité) : "Parmi les 420 départements en sciences humaines, représentant 130 campus différents, qui ont répandu à l'enquête, 16 % ont accès à SPSS, 16 % à OSIRIS, 13 % à DATA-TEXT 2e génération. En outre 87 % mentionnent d'autres packages, dont le nombre total atteint 170".

(2) principalement OSIRIS I

(3) principalement OSIRIS II