

**SÉMINFOR 1
PREMIER SÉMINAIRE INFORMATIQUE
DE L'ORSTOM**

**BASES DE DONNEES ET SYSTEMES D'INFORMATION :
QUELLES METHODES ?**

PARIS du 6 au 8 octobre 1987

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite » (alinéa 1^{er} de l'article 40).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal.

**SÉMINFOR 1
PREMIER SÉMINAIRE INFORMATIQUE
DE L'ORSTOM**

**BASES DE DONNEES ET SYSTEMES D'INFORMATION :
QUELLES METHODES ?**

PARIS du 6 au 8 octobre 1987

Textes recueillis et présentés par Patrick Séchet

Éditions de l'ORSTOM

INSTITUT FRANÇAIS DE RECHERCHE SCIENTIFIQUE POUR LE DÉVELOPPEMENT EN COOPÉRATION

Collection **COLLOQUES et SÉMINAIRES**

PARIS 1988

AVANT-PROPOS

Séminfor 1, premier séminaire informatique de l'ORSTOM, a été organisé par la Commission Scientifique des Sciences de l'Ingénieur et de la Communication et la Mission Technique Informatique. Cette réunion s'est déroulée dans l'amphithéâtre du siège de l'ORSTOM, rue La Fayette à Paris, du 06 au 08 octobre 1987 et a rassemblé plus d'une cinquantaine de participants, développeurs et utilisateurs de l'informatique, dans les disciplines les plus diverses de l'Institut.

Dans son discours introductif, Monsieur Philippe Tenneson, Directeur Général, a souligné sa satisfaction pour la réalisation de cette réunion et a brièvement évoqué l'importance que revêt actuellement l'outil informatique dans la réalisation des missions de l'ORSTOM.

Vingt et une communications, organisées en cinq sessions, ont été présentées (voir sommaire) : une large place a été faite aux débats, co-présidés par MM. Jean Déjardin, président de la Commission Scientifique 07 et Pascal Renaud, chef de la Mission Technique Informatique, des éléments les plus marquants de chaque exposé. La discussion était favorisée par la préparation préalable de questions qui ont été posées à l'auteur à l'issue de son intervention.

Un certain nombre de recommandations ont été débattues et adoptées comme conclusion de ce séminaire. En particulier, il a été suggéré de renouveler cette expérience dès l'automne 1988, en organisant un Séminfor 2 à Montpellier, sur un sujet plus centré et en l'ouvrant éventuellement à des communications extérieures à l'ORSTOM.

SOMMAIRE

Session 1 : La collecte des données

- Gérard Cochonneau* : MASQUE - un générateur/gestionnaire d'écrans utilisable par divers langages de programmation. 11
- Benoît Ferry & Pierre Cantrelle* : S.I.C. Système intégré de collecte (progiciel de collecte et d'analyse de données quantitatives sur micro-ordinateurs portables). 29
- Agnès Guillaume et Philippe Hamelin* : Discussion sur la réalisation et l'application d'un logiciel de traitement d'enquêtes. 43
- Marc Henrotte, Michel Lardy et Jeanne Tabbagh* : Télé-observation informatisée des volcans actifs. 53
- Patrick Séchet* : Intérêt, création et utilisation des grilles d'écran. 61
- Jacques Vassal* : Lecture et traitement in situ sur micro-ordinateur de cassettes de données numériques de terrain. 91
- Jacques Vaugelade et Marie Piron* : Présentation d'un gestionnaire de données numériques hiérarchisées destiné au dépouillement d'enquêtes. 97

Session 2 : La constitution de bases de données

- François Anthony* : La base de données des ressources génétiques des caféiers africains en collection en Côte d'Ivoire. 107
- Philippe Bourret* : Gestion microinformatique simplifiée des fichiers d'inventaire et leurs associés (POI). 115
- Gérard Cochonneau* : HYDROM : un logiciel de gestion des données d'un réseau d'observation hydrométrique. 119
- G.Cremers, C.Feuillet, J.J.de Granville, M.Hoff & H.Brisse* : Code floristique des ptéridophytes de Guyane française. 137
- M.Hoff, G.Cremers, C.Feuillet et J.J.de Granville* : AUBLET - la banque de données de l'herbier du centre ORSTOM de Cayenne, premiers résultats. 145
- Jean-Jacques Lechauve* : Le système de données CROISIERES. 171
- François Missègue* : GEOMER - une base de données de géophysique marine. 189
- Marc Pansu* : La gestion des données de laboratoires d'analyses physico-chimiques : exemple d'application, MIDAS. 203
- Patrick Séchet* : Méthode d'analyse structurée pour la mise au point de logiciels. 221

Session 3 : Bases de données documentaires

Patrick Gubry : POPCAM : base de données bibliographiques informatisée sur les études de population au Cameroun, évolution et perspectives. 251

Christiane Humbel : Logiciels d'application à la gestion de la bibliothèque (gestion des commandes d'ouvrages et gestion des prêts). 281

Jean-Paul Roux-Fouillet : HORIZON : base bibliographique ORSTOM, présentation. 285

Session 4 : Informations géographiques

Paul Faure et Claude Benizri : FORBAN : un logiciel infographique de cartographie thématique développé et portable sur micro-ordinateur compatible PC. 299

Philippe Waniez : SISECSO ou l'informatique au service d'une recherche en géographie humaine : la dynamique spatiale des cerrados au Brésil. 321

Session 5. Modélisation

Alain Morlière : Expérience d'utilisation d'un super-calculateur pour la mise au point d'un modèle océanographique opérationnel. 343

Session 1 :
COLLECTE DES DONNEES

MASQUE : UN GÉNÉRATEUR/GESTIONNAIRE D'ÉCRANS UTILISABLE PAR DIVERS LANGAGES DE PROGRAMMATION.

Gérard Cochonneau
(Informatique-Hydrologie)
(UR 502, Cadres Spatiaux de l'Indépendance Alimentaire)

EMBRAPA
Empresa Brasileira de Pesquisa Agropecuária
SuperCenter Venâncio 2.000
70333 - Brasília, DF (Brésil)

RESUME - Le caractère interactif des matériels informatiques disponibles depuis déjà de nombreuses années a conduit tout naturellement à développer des programmes qui communiquent avec l'opérateur, soit sous forme de questions-réponses qui se succèdent sur l'écran, soit à l'aide de grilles d'écran. Cette seconde technique, beaucoup plus performante et attrayante pour l'utilisateur apparaît vite indispensable pour la consultation ou la manipulation de données, même peu volumineuses. En contrepartie et compte tenu des ressources limitées offertes dans ce domaine par les langages de programmation, elle est assez difficile à mettre en oeuvre sans l'utilisation d'un outil spécifique.

Cette note présente un outil opérationnel sur micro-ordinateur compatible IBM-PC, qui permet :

- d'une part, de définir ou modifier des grilles d'écran ;
- d'autre part, d'utiliser ces grilles d'écran à l'intérieur de programmes pour saisir ou contrôler des données.

Conçu pour répondre aux besoins identifiés lors d'une expérience précédente sur mini-ordinateur, ce gestionnaire d'écran a d'abord été développé en langage évolué (Cobol). Les résultats satisfaisants obtenus, lors de l'utilisation de cette première version pour le développement du logiciel Hydrom, ont conduit à la mise au point d'une seconde version écrite en Assembleur Conditionnel, donc plus performante et utilisable par différents langages de programmation (Cobol, Basic compilé, Quick-Basic, Fortran-77).

L'éditeur permet de définir, consulter, imprimer les grilles d'écran ; à tout moment, lors de la définition, la grille peut être montrée à l'écran dans son état partiel. On y distingue des champs constants et des champs variables.

Le contenu des champs constants est fixé lors de la définition et chaque champ variable est caractérisé par des attributs obligatoires :

- son nom qui suffira pour l'identifier dans la grille ;
- sa position sur l'écran ;
- sa longueur ;
- le type de données qu'il accepte (alphanumérique, numérique signé ou non signé, alphabétique, etc.) ;
- et un certain nombre d'attributs optionnels pour indiquer :
 - que le champ est obligatoire lors de la saisie ;
 - que ce champ devra être totalement rempli ;
 - une valeur initiale qui sera affichée automatiquement lors de l'initialisation ;
 - des valeurs extrêmes qui ne peuvent être dépassées lors de la saisie.

A l'intérieur d'un programme, l'utilisation des grilles se fait par l'appel d'un module capable d'effectuer à la demande cinq fonctions principales :

- affichage d'une grille prédéfinie ;
- récupération de l'information saisie par l'opérateur dans un champ ou un groupe de champs variables ;
- écriture dans un champ ou un groupe de champs variables ;
- réinitialisation de la grille ou d'un groupe de champs variables ;
- émission d'un message d'erreur fourni par le programme utilisateur.

Lors de la saisie, l'opérateur peut, au lieu de remplir un champ, actionner une touche de fonction ; ce qui sera communiqué au programme principal à travers un code de retour et pourra être utilisé dans le programme principal pour déclencher certaines opérations particulières (ruptures de séquence, répétition du contenu d'un champ, etc.).

INTRODUCTION

Le caractère interactif des matériels informatiques disponibles depuis déjà de nombreuses années a conduit tout naturellement à développer des programmes qui communiquent avec l'opérateur, soit sous forme de questions-réponses qui se succèdent sur l'écran, soit à l'aide de grilles d'écran. Cette seconde technique, beaucoup plus performante et attrayante pour l'utilisateur final apparaît vite indispensable pour la consultation ou la manipulation de données, même peu volumineuses. En contrepartie, et compte tenu des ressources limitées offertes dans ce domaine par les lan-

gages de programmation, elle est assez difficile à mettre en oeuvre sans l'utilisation d'un outil spécifique.

Mis au point et utilisé à l'ORSTOM, Masque est un logiciel de génération et de gestion de grilles d'écran qui répond à ces besoins. A l'heure actuelle, il est opérationnel pour les langages de programmation suivants :

- Basic Compilé de MicroSoft ;
- Quick Basic de MicroSoft ;
- Level-II Cobol de MicroFocus ;
- Fortran-77 de MicroSoft ;
- TurboPascal de Borland.

1. DESCRIPTION GENERALE

Une grille d'écran se présente à l'utilisateur comme un partage de l'écran en zones constantes dont le contenu est prédéfini et en zones variables réservées à l'entrée de données qui sont alors récupérées par le programme d'application qui utilise la grille d'écran. Cette idée de "récupération" des données est d'ailleurs souvent artificielle, dans la mesure où, généralement, l'application ne lit pas les données dans la zone variable mais les mémorise au fur et à mesure de leur digitation et se sert parallèlement de la zone variable pour informer l'utilisateur de ce qu'il a digité. Une grille est donc comparable à un formulaire comme on en rencontre dans la vie courante (déclaration d'impôts, fiche de renseignements etc...). A ce propos, lorsqu'un système utilise des grilles d'écran pour saisir des formulaires préalablement remplis, il est très important de concevoir des grilles qui ressemblent le plus possible aux formulaires, afin de faciliter les opérations de saisie et de minimiser les risques de confusion ou d'oubli d'informations. On emploie également le terme "masque d'écran" au lieu de grille, l'ensemble des champs constants pouvant être vu comme un masque collé sur l'écran, la partie dynamique de l'écran se réduisant alors aux champs variables (les trous du masque). Nous ne traiterons pas séparément les menus qui seront considérés comme un cas particulier de grille avec un seul champ variable (le numéro de l'option à choisir) et autant de champs constants que d'options offertes. Des menus plus professionnels peuvent être élaborés en définissant un champ variable devant chaque option et en déplaçant une flèche d'un champ à l'autre pour faire son choix.

Le logiciel Masque se compose de deux parties distinctes et indépendantes (Figure 1) :

- un générateur de grilles d'écran qui permet de définir des grilles, de les modifier, de les imprimer, de les tester ;

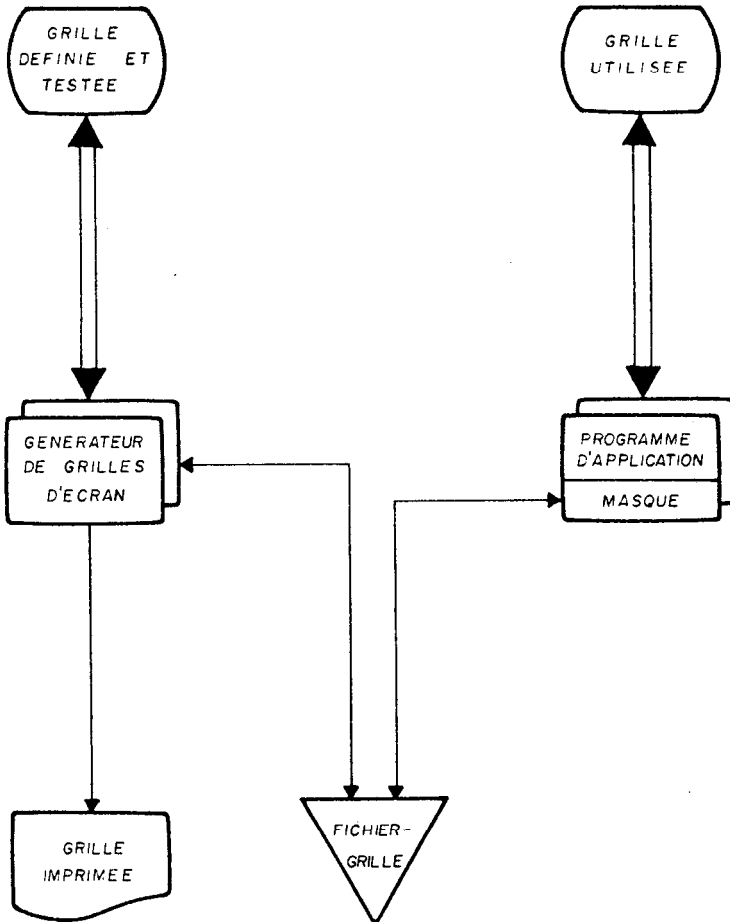


Figure 1

- un module appelé Masque qui, intégré dans un programme d'application, permet d'utiliser ces grilles.

Pour le générateur de grilles et pour le programme d'application, une grille est avant tout un fichier que nous appellerons d'appeler dans toute la suite : fichier-grille. Ce fichier est indépendant du langage de programmation dans lequel sera écrit le programme qui l'utilisera.

Pour le module Masque, une grille est un ensemble d'informations lues dans un fichier, et qui sont nécessaires et suffisantes pour décrire les zones constantes et surtout les champs variables de la grille.

2. DESCRIPTION DES GRILLES D'ECRAN

Un *champ constant* est bien entendu le plus simple : il est parfaitement défini par son contenu et ses coordonnées sur l'écran, exprimées en numéro-de-ligne, numéro-de-colonne. Il peut représenter un titre, des informations destinées à l'utilisateur mais plus généralement, il indique la signification d'un champ variable auquel il est associé.

Un *champ variable* est caractérisé par plusieurs paramètres, obligatoires ou optionnels et identifié par un nom de un à trois caractères qui est utilisé par le programme d'application pour communiquer avec la grille.

L'utilisation d'un nom pour identifier le champ présente plusieurs avantages. Tout d'abord, cela clarifie les relations entre le programme d'application et la grille, si l'on prend soin de nommer les champs en rapport avec leur signification. D'autre part, l'ajout ou la suppression d'un champ sur la grille se traduit de manière relativement souple pour le programme d'application : il suffit d'y ajouter ou d'y supprimer le traitement du champ en question sans intervenir sur les autres champs ; il en est de même dans le cas d'un changement de position du champ sur la grille. Ce ne serait pas le cas si les champs étaient simplement identifiés par un numéro d'ordre lié à leur position sur l'écran, par exemple.

La description du champ variable détermine en fait le type de données qu'il acceptera lors de la saisie. Notons que cette description n'est pas globale mais qu'à chaque position du champ est associé un type de données ; ce qui permet de mélanger, à l'intérieur d'un même champ, des types différents tout en bénéficiant d'un contrôle maximum lors de la saisie. Prenons par exemple le cas d'un champ de huit positions dont les quatre premières doivent être des lettres et les quatre dernières, des chiffres : une description globale obligerait à définir un champ alphanumérique et à contrôler la valeur saisie dans le programme

d'application ; alors qu'une description position par position permet au module Masque d'effectuer un contrôle immédiat et d'interdire l'entrée d'une lettre dans la partie numérique et inversement. Les types de données possibles sont les suivants :

- A : type alphabétique (majuscules ou minuscules non accentuées) ;
- X : type alphanumérique (chiffres de 0 à 9, majuscules, minuscules accentuées ou non) ;
- D : type numérique (chiffres de 0 à 9) ;
- S : type signe (+ ou -) ;
- E : type ASCII étendu (n'importe quel caractère dont le code est supérieur ou égal à 31 dans la table ASCII).

Tout autre caractère utilisé dans la définition du champ sera considéré comme un délimiteur et affiché automatiquement sur l'écran mais, comme un champ constant, il ne sera pas accessible à l'utilisateur lors de la saisie. Les caractères "," et "." sont des délimiteurs particuliers utiles pour séparer les parties entières et décimales des champs en virgule fixe.

Une facilité supplémentaire, semblable à ce qui existe dans la clause Picture du Cobol, est offerte pour répéter le même type de données, ou le même délimiteur, sur des positions consécutives. La description du champ pris comme exemple plus haut peut s'écrire indifféremment AAAADDDD ou A(4)D(4).

La combinaison de ces types de données et des délimiteurs permet ainsi de définir des champs simples (numériques, alphabétiques...) ou des champs composés (dates, téléphones, numérique signé avec ou sans point décimal etc...).

Le dernier paramètre obligatoire pour définir pleinement un champ variable est la position à laquelle il commence sur l'écran exprimée, comme pour les champs constants, par un numéro de ligne et un numéro de colonne.

Remarquons que la longueur du champ n'est pas un paramètre nécessaire puisqu'elle est implicitement définie par la description position par position du type de données. Toutefois, un champ ne peut avoir une longueur totale supérieure à 80 et doit se trouver entièrement sur la même ligne.

Pour compléter la description d'un champ variable, des paramètres optionnels sont disponibles :

- un attribut qui rend le champ obligatoire lors de l'utilisation de la grille, l'opérateur ne pourra alors passer ce champ sans y introduire une valeur en accord avec le type de données qu'il peut recevoir ;

- un attribut qui, lors de l'utilisation de la grille, interdira à l'opérateur de passer tant que le champ ne sera pas rempli totalement ;

- une valeur initiale avec laquelle le champ sera rempli dès l'affichage de la grille (cette valeur sera bien sûr modifiable par l'opérateur lors de la saisie ;

- un intervalle des valeurs que peut prendre le champ, défini par une valeur minimale et une valeur maximale ; ce paramètre introduit un contrôle supplémentaire au niveau du module Masque, contrôle qui est ainsi économisé dans le programme d'application.

Ces valeurs minimales et maximales ne sont autorisées que si la longueur totale du champ ne dépasse pas cinq. Malgré ce facteur limitant, ces paramètres sont intéressants pour des champs qui correspondent à un numéro de mois, à une option, etc..

Si le nombre et la taille des champs fixes ne sont limités que par la taille de l'écran (24 lignes de 80 colonnes), un certain nombre de limites en relation avec les champs variables sont imposées par le logiciel. C'est ainsi que, pour une même grille le nombre de champs doit rester inférieur à 220 et le total des zones variables ne peut dépasser 1200 positions soit 62% de la surface totale de l'écran. Ces valeurs maximales ont été choisies suffisamment élevées pour ne pas représenter une contrainte très sévère.

3. LE GENERATEUR D'ECRANS

Ecrit en Turbo-Pascal, le générateur de grilles d'écran se présente comme un module exécutable capable de réaliser cinq fonctions que nous allons décrire dans ce chapitre. Il utilise lui-même le module Masque pour communiquer avec l'utilisateur. Il est indispensable, lors du développement de l'application, pour définir les grilles, mais il est inutile de le fournir à l'utilisateur final.

3.1. Visualiser une grille

Il s'agit tout simplement de faire apparaître sur l'écran une grille déjà enregistrée dans un fichier-grille. Cette présentation de la grille est identique à ce qu'elle sera lors de son utilisation dans un programme d'application.

3.2. Manipuler une grille

C'est la fonction la plus importante offerte par le générateur puisqu'elle permet de définir et corriger de manière interactive des grilles d'écran avant de les conserver dans des fichier-grilles. Pour décrire chaque champ variable, l'utilisateur doit remplir une grille d'écran avec tous les paramètres obligatoires et éventuellement des paramètres optionnels. Pour décrire les champs constants, deux manières peuvent être employées : procéder comme pour les champs variables ou définir le masque directement en plaçant le

contenu des zones constantes à leur position voulue sur un écran vierge. Les mêmes facilités sont disponibles pour corriger ou supprimer un champ. Le logiciel effectue au fur et à mesure les contrôles nécessaires pour vérifier que la description est conforme aux règles exposées plus haut. Enfin, à tout moment, l'utilisateur peut afficher sur l'écran l'état de la grille en cours de définition et la sauver dans un fichier-grille avec possibilité d'en changer le nom ou le répertoire.

3.3. Imprimer une grille

La figure 2 montre un exemple de listing produit par cette fonction. Le logiciel lit la grille à imprimer dans un fichier-grille et produit une impression en deux parties : la grille telle qu'elle sera affichée, placée entre deux gabarits de numéros de colonnes et un tableau donnant les caractéristiques des champs variables. Pour chacun des champs variables, sont consignés dans ce tableau, le nom du champ, sa position, sa description, ses attributs (M : champ obligatoire, T : champ à remplir complètement, I : champ avec valeur initiale), ses valeurs initiales, minimales et maximales.

Ce listing peut être archivé ou utilisé pour contrôler la définition ou comme point de départ pour définir une autre grille, mais il est aussi indispensable lors de la définition du programme qui va utiliser la grille.

3.4. Tester une grille

L'utilisateur peut profiter de cette fonction pour simuler un emploi réel de la grille qu'il a défini et sauvegardé. Cela lui permet, avant même d'écrire son programme d'application, de vérifier la fonctionnalité de la grille et la bonne définition des champs variables (type de données, valeurs initiale, minimale, maximale, longueur de champ). La saisie se fait dans l'ordre dans lequel ont été définis les champs ; cet ordre est en général différent de celui que l'utilisateur s'imposera dans son programme d'application. Les valeurs saisies ne sont bien entendu pas conservées.

3.5. Compiler une grille

Bien que redondante avec la fonction de manipulation de grilles, cette fonction a quand même été développée. Elle transforme une description de grille saisie dans un autre éditeur de texte en fichier-grille compatible avec le logiciel Masque. Un utilisateur occasionnel peut ainsi éviter d'apprendre à se servir de la fonction de manipulation et définir sa grille à l'aide d'un éditeur de texte qui lui est plus familier. Le fichier ainsi saisi, que nous conviendrons de nommer fichier-source doit cependant respecter un format strict décrit dans la notice d'utilisation : un enregistrement pour chaque

ORSTOM - Masque Définition de la grille DEMO.GRC
Page 1

...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8
ORSTOM LOGICIEL "MASQUE"

EXEMPLE D'UTILISATION

Nom : Né(e) le : .././....
Prénom : Indice : ...

Affectation : MET (MET/O-M)
Téléphone : `..\$-..-..

Année :
Mois : ..
Salaire mensuel brut : FF
Salaire mensuel net : FF

Validation : ... (OUI/NON)

Aide aux touches de fonction:

- F1 : retour en arrière
- F5 : duplication d'un champ
- F3 : réinitialisation
- SF1: fin de saisie

...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...8

ORSTOM - Masque Définition de la grille DEMO.GRC Page 2

NO	Nom	Lig	Col	Description	Attributs	Val. Ini	Val. Min	Val. Max
1	NOM	05	12	X(30)	M			
2	PRE	06	12	X(30)	M			
3	D	05	61	DD/DD/DDDD	T			
4	IND	06	61	DDD	T			
5	AF	08	16	XXX	M T I	MET		
6	TEL	09	16	'DD\$-DD-DD-DD	T			
7	A	12	11	DDDD	M T			
8	M	13	11	DD	M T		01	12
9	SB	14	25	SD(5),DD	M			
10	SN	15	25	SD(5),DD	M			
11	VAL	17	15	AAA	M T			

Fig. 2 : exemple d'impression de grille.

zone, constante ou variable, contient tous les paramètres nécessaires à la définition du champ. Dès que le fichier source a été compilé avec succès, un fichier-grille est créé et toutes les fonctions du générateur d'écran peuvent être appliquées.

4. LE MODULE MASQUE

Comme nous l'avons résumé plus haut, ce module est dédié à l'utilisation d'une grille déjà définie, dans un programme d'application. Le programme le voit comme un module externe (subroutine ou procédure, selon le langage) et communique avec lui par des paramètres en nombre variable selon la fonction demandée qui est elle-même un des paramètres. Masque réalise *sept fonctions* et dans tous les cas, au moins deux paramètres sont nécessaires : l'un contient la fonction que l'on demande à Masque de réaliser, l'autre est un code de retour. Ce dernier doit être testé au retour de Masque pour détecter si une erreur s'est produite lors de la réalisation de la fonction demandée et identifier cette erreur. Parmi les erreurs possibles, citons le cas où le fichier-grille n'a pas pu être accédé ou est non conforme et le cas où le nom d'un champ communiqué à Masque ne correspond pas à un champ de la grille. Ces erreurs se produisent le plus souvent lors de la mise au point du programme, mais le code de retour joue un autre rôle très important lorsque le programme est opérationnel comme nous le verrons dans la description de la fonction de saisie. La description de chacune de ces fonctions est illustrée par un exemple, en langage Basic et en langage Cobol, basé sur la grille imprimée en figure 2.

La *fonction Get* doit être exécutée avant toute opération à réaliser sur la grille : elle charge la description de la grille à partir du fichier-grille et l'affiche dans son état initial. Le nom du fichier-grille doit être fourni lors de l'appel au module Masque. Chaque position des champs variables est matérialisée par un point, les délimiteurs sont affichés ainsi que les valeurs initiales des champs initialisés.

Basic :

```

10 GRILLE$ = "DEMO."           `GRILLE:nom du fichier-grille
20 OP$      = SPACE$(2)        `OP      :code de l'opération
30 STAT$    = SPACE$(2)        `STAT    :code de retour
...
40 OP$ = "GT"                  `GT      :opération Get
50 CALL MASQUE(GRILLE$,STAT$,OP$) `appel de Masque pour
                                `afficher la grille

```

Cobol :

```
01 GRILLE PIC X(5) VALUE 'DEMO.'.
01 OP      PIC XX.
01 STAT   PIC XX.
...
MOVE 'GT' TO OP.
CALL 'MASQUE' USING GRILLE STAT OP.
```

La *fonction Read* est la plus souvent utilisée et la plus sophistiquée : elle autorise l'opérateur à saisir des valeurs dans un ou plusieurs champs variables. La liste des noms des champs à saisir doit être communiquée lors de l'appel de Masque ainsi que l'adresse d'une zone qui contiendra, au retour, les valeurs saisies par l'opérateur. Un effort particulier a été fait pour optimiser les performances de cette fonction. Chaque touche actionnée est contrôlée en accord avec le type de données que peut recevoir la position où se trouve le curseur et toute valeur non conforme est refusée et provoque l'émission d'un signal d'alarme. Les positions qui correspondent à des délimiteurs sont passées automatiquement puisqu'elles sont déjà remplies. Les touches de déplacement du curseur vers la gauche ou vers la droite sont utilisables à l'intérieur d'un champ ainsi que le mode insertion et la touche de suppression de caractères ; les caractères qui, éventuellement, déborderaient du champ en mode insertion disparaissent de l'écran mais restent accessibles si d'autres sont supprimés, à condition bien sûr de rester dans le même champ.

La saisie d'un champ est considérée comme terminée si l'opérateur vient d'informer la dernière position du champ ou si la touche Retour-Chariot a été actionnée et que l'état du champ satisfait les conditions imposées par les attributs de champ obligatoire ou à remplir complètement. Dans le cas d'un champ numérique, le logiciel effectue un cadrage automatique à droite et complète le champ avec des zéros non significatifs mais en tenant compte du signe et du point décimal. Par contre, un champ alphabétique ou alpha-numérique est cadré à gauche.

Le code de retour joue un rôle important dans cette fonction de saisie car si l'opérateur actionne une des 47 touches de fonction autorisées (F1 à F40, Esc, etc.) une valeur est renvoyée dans ce code de retour qui doit être testé dans le programme appelant pour faire réaliser une opération particulière telle que l'appel d'un écran d'aide, le passage à un autre champ, la duplication d'une valeur antérieure dans un champ, la sortie du programme etc..

Basic :

```

100 C.DATE$ = "D."           `C.DATE:nom du champ à saisir
110 V.DATE$ = SPACES(10)    `V.DATE:pour recevoir la date
                               `saisie
...
120 OP$ = "RD"              ` RD   :opération Read
130 CALL MASQUE(C.DATE$,V.DATE$,STAT$,OP$) `appel de Masque
                               `pour saisir le champ D
140 IF STAT$ <> "00" THEN traiter_touche_de_fonction.

```

Cobol :

```

01 C-DATE PIC XX VALUE 'D.'.
01 V-DATE PIC X(10).
...
MOVE 'RD' TO OP.
CALL 'MASQUE' USING C-DATE V-DATE STAT OP.
IF STAT NOT = ZERO THEN traiter_touche_de_fonction

```

La *fonction Reset* permet au programme d'application de remettre dans leur état initial un ou plusieurs champs variables de la grille ; nous entendons toujours par état initial la matérialisation de chaque position par un point ou l'affichage de la valeur initiale. Cette fonction peut par exemple être déclenchée, à la demande de l'opérateur (avec une touche de fonction), pour effacer le contenu d'un champ. La liste des noms des champs à réinitialiser doit être fournie avant l'appel du module Masque sauf dans le cas particulier où l'on souhaite remettre la grille entière dans son état initial. Le résultat est alors apparemment identique à ce que réalise la fonction Get, mais en réalité aucun accès n'est fait au fichier-grille puisque la description de la grille est déjà présente dans la mémoire.

Basic :

```

200 OP$ = "RS"              ` RS   :opération Reset
210 CALL MASQUE(C.DATE$,STAT$,OP$) `appel de Masque pour
                               `réinitialiser le champ D

```

Cobol :

```

MOVE 'RS' TO OP.
CALL 'MASQUE' USING C-DATE STAT OP.

```

L'écriture directe dans un champ, par le programme d'application se fait grâce à la *fonction Write*. La liste des noms des champs que l'on désire remplir ainsi qu'une zone qui contient les valeurs à afficher doivent être communiquées au module Masque lorsqu'il est appelé. Cette opération peut être déclenchée, à la

demande de l'opérateur, pour répéter dans un champ la valeur saisie précédemment mais aussi pour lui communiquer des informations. Son utilisation la plus fréquente correspond à l'affichage de données déjà saisies, en vue de les corriger.

Basic :

```

300 NOM$ = "AF D."           `nom des champs à aff.
310 VALEUR$ = "O-M25/12/1900" `affectation+date à afficher
...
320 OP$ = "WR"              ` WR   :opération Write
330 CALL MASQUE(NOM$,VALEUR$,STAT$,OP$) `appel de Masque pour
                                     `afficher l'affectation et
                                     `la date

```

Cobol :

```

01 NOM PIC X(6) VALUE 'AF D.'.
01 VALEUR.
   02 FILLER PIC X(03) VALUE 'O-M'.
   02 FILLER PIC X(10) VALUE '25/12/1900'.
...
MOVE 'WR' TO OP.
CALL 'MASQUE' USING NOM VALEUR STAT OP.

```

La *fonction Clear* efface tout simplement l'écran ; elle ne nécessite aucun paramètre particulier.

Basic :

```

400 OP$ = "CL"              ` CL   :opération Clear
410 CALL MASQUE(STAT$,OP$) `appel de Masque pour effacer l'écran

```

Cobol :

```

MOVE 'CL' TO OP.
CALL 'MASQUE' USING STAT OP.

```

La *fonction Display* affiche un message sur la vingt-cinquième ligne de l'écran, provoque l'émission d'un signal d'alarme et attend que l'opérateur appuie sur une touche avant d'effacer le message et de retourner au programme appelant. Le message peut être un message d'erreur ou d'information et doit être communiqué à Masque lors de l'appel.

Basic :

```

500 MSG1$ = "Valeur erronée$" `message terminé par $
...
510 OP$ = "DP" ` DP :opération Display
520 CALL MASQUE(MSG1$.STAT$.OP$) `appel de Masque pour afficher
`le message en bas de l'écran

```

Cobol :

```

77 MSG1 PIC X(15) VALUE 'Valeur erronée$'.
...
MOVE 'DP' TO OP.
CALL 'MASQUE' USING MSG1 STAT OP.

```

La fonction *Output-Color* permet l'affichage d'une information à partir d'une position quelconque de l'écran et en bénéficiant du choix des attributs d'affichage. La position sur l'écran, les attributs d'affichage (vidéo inversée, clignotement, couleur) ainsi que la chaîne de caractères à afficher doivent être communiqués à Masque lors de l'appel.

Basic :

```

600 MSG2$ = "Calculs en cours$" `message terminé par $
610 CUR$ = "2015" `ligne 20, colonne 15
620 COUL$ = "000705" `07:vidéo inversée
`05:clignotement
...
630 OP$ = "OC" ` OC :opération Output-Color
640 CALL MASQUE(MSG2$.CUR$.COUL$.STAT$.OP$) `appel de Masque
`pour afficher le message

```

Cobol :

```

77 MSG2 PIC X(17) VALUE 'Calculs en cours$'.
77 CUR PIC X(4) VALUE '2015'.
77 COUL PIC X(6) VALUE '000705'.
...
MOVE 'OC' TO OP.
CALL 'MASQUE' USING MSG2 CUR COUL STAT OP.

```

Notons que ces trois dernières fonctions peuvent être utilisées indépendamment de l'utilisation ou non d'une grille d'écran.

5. HISTORIQUE

L'idée ou plutôt la nécessité de développer ce logiciel de grilles d'écran est apparue à la suite du changement de matériel informatique causé par le transfert du Laboratoire d'Hydrologie. L'utilisation d'un tel outil sur le Mini-6 entre 1981 et 1984 ayant montré tout l'avantage que l'on pouvait en tirer, il s'avérait indispensable de continuer dans cette voie. La première application à adapter sur les micro-ordinateurs IBM-PC étant écrite en Cobol, nous avons cherché un gestionnaire d'écran adapté à nos besoins. A cette date (début 1985), de tels logiciels étaient moins nombreux qu'aujourd'hui et les deux qui furent testés, Display Manager et Forms-II n'étaient pas satisfaisants : l'un n'étant pas utilisable par le langage Cobol, l'autre étant par trop rudimentaire. La décision a donc été prise de développer un outil qui soit assez ressemblant par ses possibilités et son interface avec celui utilisé précédemment, ceci afin de faciliter l'adaptation des programmes déjà opérationnels.

Ecrite en Level-II-Cobol, comme les programmes de l'application à adapter, la première version se composait d'un module d'utilisation des grilles et d'une partie de manipulation des grilles des plus rudimentaire : utilisation d'un éditeur classique pour saisir les grilles et transformation du fichier saisi, dans un format reconnu par le module d'utilisation. Ce prototype a été utilisé pendant plus d'un an et la première version du logiciel Hydrom, diffusée au printemps 1986 à des fins de test, l'utilisait encore.

Il devenait dès lors indispensable d'améliorer les performances, notamment au niveau de la rapidité d'affichage, et d'introduire des fonctionnalités - telles que le cadrage des valeurs numériques, le mode insertion, la suppression - qu'il n'était pas viable de développer dans un langage évolué, toujours pour des raisons de rapidité d'exécution. D'autre part, certaines applications graphiques d'Hydrom étaient développées en Basic et, pour des raisons d'homogénéité, il était intéressant d'y utiliser le même logiciel de grille d'écran. Une nouvelle version du module d'utilisation des grilles a donc été écrite en Assembleur Conditionnel qui représentait la seule alternative pour, à la fois, augmenter l'efficacité et permettre l'interface avec divers langages de programmation.

Ce n'est que plus récemment que des versions du module Masque ont été développées pour le Fortran-77 et le Turbo-Pascal et qu'un générateur d'écrans plus opérationnel a été mis au point pour en arriver à un produit plus complet.

6. POINTS FORTS ET PRINCIPAUX PROBLEMES RENCONTRES

L'hétérogénéité des compilateurs disponibles pour micro-ordinateur représente un obstacle important dès lors qu'un même module écrit en assembleur doit s'interfacer avec différents langages. En effet, la communication entre un programme appelant et un module appelé est réalisée de façon particulière dans chaque compilateur et varie parfois d'une version à l'autre du même compilateur. Force est de constater aussi que, très souvent, la documentation fournie à ce sujet dans les manuels de référence se résume à un paragraphe d'informations assez imprécises et parfois erronées qu'il est indispensable de vérifier soigneusement. Certains compilateurs (Fortran-77, Basic) produisent un module objet qui nécessite une édition de liens alors que d'autres (Level-II-Cobol, Turbo-Pascal) produisent directement un module exécutable. Le module assembleur appelé doit être adapté à chaque cas. La version pour le Turbo-Pascal est la plus complexe car les adresses des variables internes au module doivent être recalculées en temps d'exécution.

Bien que les paramètres qui sont communiqués entre le programme appelant et le module aient été définis le plus universellement possible comme des chaînes de caractères, il a fallu tenir compte de la façon particulière dont est structuré ce type de variables en Basic (longueur, adresse du contenu) et en Turbo-Pascal (longueur, contenu).

Les fonctions d'affichages offertes par le système opérationnel DOS se révélant assez peu efficaces, l'édition sur l'écran se fait en écrivant directement dans la mémoire, du moins si l'on travaille en mode texte. On arrive ainsi à une vitesse d'affichage assez spectaculaire. L'utilisation du module Masque en mode graphique est également possible, mais l'affichage sur l'écran reste alors tributaire de la lenteur des fonctions du DOS.

L'absence d'un code de retour a été envisagée ; dans le cas d'une erreur détectée dans la routine Masque, on aurait pu se contenter de retourner au DOS après avoir signalé l'erreur. Mais, dans le cas du développement d'un programme en Turbo-Pascal par exemple où le test et l'édition d'un programme se réalisent simultanément, les dernières modifications du programme source auraient été perdues. On pouvait aussi imaginer d'imposer des touches de fonction dans le module Masque pour réaliser des fonctions classiques telles que le retour au champ précédent, la réinitialisation du champ etc.. et s'affranchir ainsi du code de retour utilisé dans la fonction Read pour informer le programme appelant de la touche de fonction qui a été actionnée. Mais ceci aurait entraîné la perte de toute la souplesse qui permet à l'utilisateur de

choisir ses propres touches de fonction et de définir l'opération associée.

La combinaison des fonctions Write et Reset du module Masque permet une sorte de paramétrage de la grille en fonction des informations qui y sont saisies. C'est ainsi que des zones variables peuvent être définies dans la grille, blanchies dès l'affichage initial et utilisées ensuite par couples (intitulé de champ, champ lui-même) pour saisir des informations en rapport avec la valeur donnée par l'opérateur à un autre champ.

Si l'on se réfère aux grilles d'écran habituelles, on peut penser que la possibilité de définir 220 champs variables est exagérée. Cependant cette facilité est parfois intéressante, comme ce fut le cas dans une application peu ordinaire du module Masque : son utilisation dans un programme de saisie sur une table à digitaliser. Le clavier, désactivé, était remplacé par un menu dessiné sur la table et la grille ne servait qu'à visualiser les données saisies et à informer l'opérateur. Mais pour accompagner la saisie faite sur le menu, chaque position des zones variables devait elle-même être isolée et représentée par un champ, afin qu'à chaque saisie de coordonnées dans la zone de menu soit affiché le caractère correspondant ou réalisée la fonction demandée (effacement d'un caractère, retour en arrière, etc...). La possibilité offerte par le logiciel de définir des champs variables contigus était également indispensable.

Il n'est pas courant de réaliser à l'ORSTOM ce genre de travail dans la mesure où l'on n'a pas vocation de développer des outils informatiques d'intérêt général mais plutôt des applications. Comme on a pu le voir, la partie générateur d'écrans a, depuis le début, été défavorisée par rapport au module d'utilisation des grilles qui est le plus important pour l'utilisateur final de l'application. De même, le manque de documentation du logiciel, laissée de côté jusqu'au dernier moment en a réduit l'emploi : mises à part quelques utilisations marginales, seules deux applications, Hydrom et Pluviom, en font usage. Il s'agit toutefois d'applications de grande envergure, Hydrom notamment utilise 62 masques d'écran dont 13 menus.

S.I.C. : SYSTEME INTEGRE DE COLLECTE

PROGICIEL DE COLLECTE ET D'ANALYSE DE DONNÉES QUANTITATIVES SUR MICRO-ORDINATEURS PORTABLES

Benoît Ferry et Pierre Cantrelle
(Démographie)
(UR 709, Population-Santé)

ORSTOM - ISD U.R. Population Santé
15-21 rue de l'Ecole de Médecine
75270 Paris Cedex 06 (France)

RESUME - *Les objectifs* : Il s'agit d'un progiciel intégré utilisable par des non-informaticiens pour mener des enquêtes et des collectes quantitatives depuis la conception jusqu'aux tableaux de résultats en s'appuyant sur une machine centrale et des machines de terrain portables.

Vise les enquêtes par interviews, y compris les plus complexes, mais peut s'appliquer également à des saisies de données et de documents sur le terrain.

Il concerne plusieurs disciplines et divers secteurs d'activité et s'applique aux pays développés comme aux pays en développement.

Le cahier global des charges :

- s'appuyer sur du matériel disponible, si possible peu cher et peu sophistiqué ;
- faire l'analyse de façon empirique à partir des tâches des collecteurs et du concepteur plutôt que de la théorie des questionnaires ;
- possibilité de génération d'un questionnaire ou d'un formulaire par un directeur d'enquête non-informaticien ;
- collecte possible sur le terrain par un personnel peu qualifié qui manipule une "machine d'enquête" intelligente mais transparente. Il doit disposer d'un ensemble d'outils qui lui permette de faire la passation du questionnaire et d'obtenir sur le terrain-même des données contrôlées et propres pour traitement sans délais ;
- portabilité des logiciels sur différents types de machine ;
- écriture de façon modulaire en vue d'améliorations et de fonctionnalités complémentaires ;

- autonomie de longue durée pour les enquêteurs sur le terrain (supérieure à huit jours sans déchargement) ;
- possibilité de collecter des informations dans des milieux sévères ;
- capacité technique des machines à se connecter éventuellement par téléphone avec un site central.

Fonctionnalités générales de l'outil :

Elles permettent la collecte de données de toutes sortes dans des champs d'applications très variés en améliorant sensiblement la qualité des données, la rapidité et le coût des opérations de collecte et de traitement de données quantitatives.

Après la génération d'un questionnaire ou d'un formulaire sur le PC grâce à un éditeur interactif qui demande de définir toutes les caractéristiques des questions et sous-questionnaires, y compris les messages et contrôles désirés, un pseudo-code peut être téléchargé sur les machines de terrain. Il permet au collecteur une saisie directe intelligente et contrôlée sur le terrain-même qui produit des enregistrements codés et propres. Après retour par téléchargement de ces données et reformatage automatique, les traitements statistiques peuvent être entrepris sur PC ou site plus important.

Le progiciel S.I.C. est écrit en TURBO-PASCAL et fonctionne sur micro-ordinateurs PC et compatibles pour la machine centrale, et sur HUSKY ou EPSON ou portable MS-DOS compatible PC pour la machine de terrain.

Ce progiciel est accompagné d'une documentation technique et d'un système de protection. Il sera commercialisé par une société privée. La maintenance, la formation et l'assistance technique seront assurées par cette société.

Il a été d'ores et déjà expérimenté dans plusieurs enquêtes en vraie grandeur.

Principales leçons des travaux effectués :

- faisabilité d'enquêtes sur des machines de terrain et de l'intégration des différentes phases des enquêtes autour de micro-ordinateurs ;
- la formation, les coûts, les connexions entre machines ne constituent pas des handicaps importants ;
- nécessité pour l'analyse de partir des tâches du concepteur et de l'enquêteur ;
- aller profondément dans l'analyse pour la correction des erreurs et incohérences pour offrir à l'enquêteur un ensemble d'outils puissants mais simples d'accès ;

- l'entrée dans une série de logiciels classiques : dBASE III, SPSS-PC, SAS, ITEM, STATGRAPHICS, etc., ne pose pas de problèmes particuliers ;

- il s'avère que cette approche permet un gain substantiel pour la qualité des données, la rapidité et le coût ;

- il sera facile d'envisager une machine intermédiaire dédiée à la supervision sur le terrain ;

- dans les domaines qui sont les nôtres, notre équipe est considérée comme pionnière tant en France qu'aux Etats-Unis et par divers organismes internationaux.

Le progiciel S.I.C. a été développé au sein de l'UR Population-Santé de l'ORSTOM en collaboration avec différents intervenants privés. Sa valorisation est en cours actuellement auprès d'une société privée qui va en assurer le développement la maintenance et la commercialisation.

AVANT-PROPOS

Le progiciel S.I.C. a été développé au sein de l'Unité de Recherche Population-Santé de l'ORSTOM en collaboration avec différents intervenants individuels et privés. Il a eu le soutien de la DIVA de l'ORSTOM et a bénéficié de crédits incitatifs. L'Agence pour l'Informatique (ADI) a également apporté son appui.

Sa valorisation est en cours actuellement auprès d'une société privée qui en assurera le développement, la maintenance et la commercialisation.

1. PRESENTATION GENERALE DU PROBLEME

Les enquêtes quantitatives se décomposent en quatre phases : la conception, la collecte d'information sur le terrain, l'exploitation et l'analyse.

De manière générale la collecte se fait sur questionnaire papier. Après les opérations de chiffrage et de saisie-vérification, l'exploitation et les contrôles de cohérence se faisaient sur gros ordinateurs, et plus récemment, en partie, sur micro-ordinateur.

A l'heure actuelle on s'efforce de rapprocher au maximum l'exploitation de la collecte sur le terrain. C'est ainsi que se sont développés les logiciels de saisie contrôlée. Il s'agit de saisie de données et de contrôle après le terrain.

Cette phase de chiffrage, saisie, contrôles divers et dépouillement peut s'avérer très longue, voire pour certaines enquêtes, décourager tout projet d'analyse des données collectées, autre qu'une analyse sommaire. Certaines enquêtes ne dépassent même

pas le stade de liasse de questionnaires sans exploitation. Dans le cas contraire, ce sont au mieux plusieurs mois, plus généralement plusieurs années qui s'écoulent entre la collecte et la rédaction des résultats.

L'apparition de micro-ordinateurs portables et de faible consommation permet d'envisager une évolution rapide des méthodes de collecte et de traitement d'enquête.

Les opérations intermédiaires (codification, saisie, contrôle), souvent difficiles à mettre en oeuvre et à coordonner, pourraient en effet être envisagées dans un système intégré qui, combinées à la collecte sur le terrain même, permettrait d'accroître considérablement et la rapidité du processus d'enquête et la qualité des données.

Il ne s'agirait plus de collecter les données sur un questionnaire papier, mais sur un micro-ordinateur de terrain portable. La qualité de la collecte peut être sensiblement améliorée en incluant des contrôles de cohérence dès le terrain sur ce micro-ordinateur.

Ainsi la collecte, la saisie et les contrôles se trouvent intégrés dès le terrain. En organisant les machines de terrain autour d'une machine centrale, les opérations d'enquête consistent alors en collecte contrôlée, transfert et tabulation-analyse et non plus en collecte, codification, saisie, contrôle et tabulation-analyse.

Notre problème étant de travailler en milieu sévère avec du personnel non qualifié et souvent peu conscient de la fragilité d'un matériel informatique, les travaux menés ici privilégient des machines de conception plus simples, mais particulièrement résistantes et de grande capacité de mémoire interne.

Le principe est alors de décentraliser au maximum la saisie sur ces petites machines et les rapprocher du lieu de création des documents, parfois sur le terrain-même.

Cette approche n'a d'intérêt que si ces machines sont reliées de temps en temps à un site central, qui peut être un ordinateur de type PC, pour le déchargement des machines, le contrôle global du travail effectué, la gestion des données et les traitements statistiques.

Dans plusieurs autres domaines que la saisie de données statistiques, la démarche de décentralisation vers le terrain a été menée et a conduit aux mêmes stratégies matérielles et organisationnelles.

On peut citer en particulier dans le domaine médical, l'aide au diagnostic. S'effectuant au début sur de gros sites informatiques, on peut voir à présent de petites machines de terrain dans les mains d'agent de santé en milieu tropical aidant au diagnostic des malades sur le terrain-même. Ce sont entre autres les expériences avec des HUSKY et des BLAISE faites par le Centre Mondial Informatique et Médecins Sans Frontières.

Pour des relevés de mesures sur le terrain et en instrumentation dans des milieux difficiles, la même démarche a été menée : collecte sur de petits micros tous terrains et connexion à des ordinateurs plus gros.

2. LE SAVOIR FAIRE SOUS-JACENT

De façon sommaire le savoir faire présent au départ pourrait se résumer ainsi : connaissance du terrain et connaissances informatiques. Ceci s'appuyant sur une équipe pluridisciplinaire.

Deux compétences majeures étaient requises : conception supervision des enquêtes et une bonne connaissance du travail du collecteur-enquêteur sur le terrain.

Dans les différentes phases des opérations les savoir-faire suivants étaient rassemblés :

- conception, réalisation d'enquêtes dans les PED en démographie, épidémiologie, nutrition, sociologie et économie et autres disciplines connexes ;

- traitement et analyse de sources diverses telles que : états-civils, études de production, budget-temps, comptes économiques ;

- réalisation de systèmes paramétrables de saisie informatisée sur micro ordinateur. Développement de logiciels de tabulation et d'analyse statistique de données d'enquêtes.

3. OBJECTIFS

Il s'agit d'un progiciel intégré utilisable par des non-informaticiens pour mener des enquêtes et des collectes quantitatives depuis la conception jusqu'aux tableaux de résultats en s'appuyant sur une machine centrale et des machines de terrain portables.

Il concerne plusieurs disciplines et divers secteurs d'activité, tels que démographie, santé, enquêtes budget-consommation, agricoles ou études de marché, et s'applique aux pays développés comme aux pays en développement.

Il vise les enquêtes par interviews, y compris les plus complexes, mais peut s'appliquer également à des saisies de données et de documents sur le terrain.

Ce système supporte l'ensemble des opérations nécessaires à une enquête : de la collecte sur le terrain réalisé à travers des micro-ordinateurs (appelés par la suite machines de terrain : MT) portables jusqu'à la gestion des données et à leur traitement réalisé sur un ordinateur plus important (appelé par la suite machine centrale : MC).

Le projet s'appuie sur des machines déjà existantes et vise à développer les logiciels qui permettront d'intégrer les différentes tâches liées à la collecte : génération de questionnaires machine, recueil sur le terrain directement sur les machines, incluant l'assistance à la passation et des contrôles de cohérence en temps réels, utilitaire de transfert sur machine centrale, gestion des données et traitements.

Ce progiciel doit déborder le domaine strict des enquêtes par interviews pour intégrer également d'autres types de saisie.

Enfin il doit déboucher sur une valorisation commerciale.

4. LE CAHIER GLOBAL DES CHARGES

- S'appuyer sur du matériel disponible, si possible peu cher et peu sophistiqué ;

- faire l'analyse de façon empirique à partir des tâches des collecteurs et du concepteur plutôt que de la théorie des questionnaires ;

- possibilité de génération d'un questionnaire ou d'un formulaire par un responsable d'enquête non-informaticien ;

- collecte possible sur le terrain par un personnel peu qualifié qui manipule une "*machine d'enquête*" intelligente mais transparente. Il doit disposer d'un ensemble d'outils qui lui permette de faire la passation du questionnaire et d'obtenir sur le terrain-même des données contrôlées et propres pour traitement sans délais ;

- portabilité des logiciels sur différents types de machine ;

- écriture de façon modulaire en vue d'améliorations et de fonctionnalités complémentaires ;

- autonomie de longue durée pour les enquêteurs sur le terrain (supérieure à huit jours sans déchargement) ;

- possibilité de collecter des informations dans des milieux sévères ;

- capacité technique des machines à se connecter éventuellement par téléphone ou d'autres moyens avec un site central.

5. L'ARCHITECTURE DU LOGICIEL ET LES DEVELOPPEMENTS

Sur la machine centrale :

- un logiciel de préparation de questionnaire : le GENERATEUR ;
- un logiciel de TRANSMISSION DES DONNEES, de formatage et de PREPARATION AU TRAITEMENT.

Sur la machine de terrain :

- un logiciel de passation de questionnaire : l'INTERPRETEUR.

5.1. Le générateur

Introduction aux concepts

- Question :
 - nom de variable
 - type de variable
 - conditions d'entrée dans la question
 - conditions d'évaluation
- Sous-questionnaire (SQ)
 - nom
 - conditions d'entrée
 - conditions de nouvelle administration (retour)

Il est sur la machine centrale (Compatible PC).

Fonctionnalités du générateur

- génération interactive de questionnaires
- définition des groupes de questions, des questions et du dictionnaire des variables selon un Langage de Description de Questionnaire (LDQ).

Réalisation sur la machine

Edition par interface écran des caractéristiques du questionnaire puis compilation et production d'un code chargeable sur la machine de terrain.

Il contient deux parties : l'éditeur et le compilateur.

a) Editeur

- il permet la saisie linéaire d'un sous-questionnaire ;
- l'ouverture d'un sous-questionnaire : la saisie des champs (LDQ) et la génération d'une liste de commandes (record) pour chaque sous-questionnaire ;
- la saisie d'une question est précédée par la définition d'un SQ (chaque question appartient à un seul SQ) ;

- la définition des champs LDQ selon le format Question. Résultats de la saisie : génération d'un record pour chaque questionnaire et création d'un dictionnaire des variables ;

- la fin du sous-questionnaire (une opération appelée du menu principal) provoque le retour au niveau plus un et ainsi permet la continuation du sous-questionnaire du niveau supérieur ;

- la fin du sous-questionnaire dominant qui correspond à la fin du questionnaire. Actions : vérification des enregistrements et éventuellement retour à la saisie ;

- fonctionnalités supplémentaires : l'édition d'état du questionnaire sous forme d'un dictionnaire ou de l'architecture et la visualisation de l'arborescence du questionnaire ;

- EDITEUR crée plusieurs fichiers dont trois seront exploités par le compilateur.

b) **Compilateur**

Il traite les fichiers générés par l'éditeur en vue de la génération d'un fichier paramètres pour la machine de terrain.

5.2. **L'interpréteur**

Il est installé sur la machine de terrain (machines sous CP/M ou MS-DOS).

Logiciel développé : Ce logiciel est chargé dans le machine de terrain avec le fichier paramètres (correspondant au questionnaire préalablement saisi).

L'interpréteur fonctionne selon le fichier paramètres.

Fonctionnalités de l'interpréteur :

- exécute pas à pas les instructions du LDQ ;
- administre les questions en tenant compte des filtres ;
- stocke les questionnaires en fichier.

5.3. **Le logiciel de préparation au traitement**

Fonctionnalités :

- transmission des données ;
- formatage des données suivant le dictionnaire des variables ;
- préparation des données pour l'entrée dans une base de données (dBASE III ou autres) et dans des logiciels statistiques de traitement de données (SPSS, SAS, BMDP, Statgraphics, etc.).

5.4. Langage choisi : Pascal

Editeur/Compilateur et Interpréteur : Turbo-Pascal Version 3.0 (Borland Inc.), langage structuré disponible sur les deux types de machine (MS-DOS et CP/M), ce qui permet une grande portabilité du logiciel.

Taille globale du logiciel : 125 K compilés.

6. FONCTIONNALITES GENERALES DE L'OUTIL ET SON UTILISATION

Les fonctionnalités générales de l'outil permettent la collecte de données de toutes sortes dans des champs d'applications très variés en améliorant sensiblement la qualité des données, la rapidité et le coût des opérations de collecte et de traitement de données quantitatives.

Après la génération d'un questionnaire ou d'un formulaire sur le PC grâce à un éditeur interactif qui demande de définir toutes les caractéristiques des questions et sous-questionnaires, y compris les messages et contrôles désirés, un pseudo-code peut être téléchargé sur les machines de terrain. Il permet au collecteur une saisie directe intelligente et contrôlée sur le terrain-même qui produit des enregistrements codés et propres. Après retour par téléchargement de ces données et reformatage automatique, les traitements statistiques peuvent être entrepris sur PC ou site plus important.

7. LES EXPERIMENTATIONS EFFECTUEES

L'ensemble d'une telle approche a été expérimentée à tous les niveaux pour sa mise au point et en assurer sa validité.

Plusieurs groupes d'expérimentation-mise au point ont été effectués.

En bureau

- pour se familiariser avec les spécificités des machines et en évaluer les possibilités ;
- pour examiner, mettre au point et expérimenter les adaptations à envisager au niveau du clavier ;
- développer différents programmes de saisie et les essayer avec du personnel ;
- expérimenter et mettre au point la formation à élaborer pour faire exécuter les tâches par du personnel non qualifié ;
- mettre au point et expérimenter les routines de début et de fin de session, de transfert des données.

A Dakar : Saisie de registres d'état-civil

Il a paru intéressant d'expérimenter ces premières approches par des opérations de saisie en situation.

Plusieurs séries d'expérimentations ont été menées dans les centres d'état-civil de Dakar en collaboration avec la Direction de la Statistique du Sénégal :

- mairie du 1er arrondissement, Centre principal d'état-civil ;

- Hôpital de Fann.

Le but était de mettre en oeuvre l'ensemble de la démarche et d'en évaluer les possibilités et les limites.

A Dakar : En enquête simple sur le terrain

L'expérimentation a eu lieu avec le personnel enquêteur de l'ORSTOM au Sénégal dans la grande banlieue de Dakar à Guedjawaye. Des interviews ont été menés dans des concessions, d'une part à partir d'un questionnaire papier et d'autre part avec les HUSKY à des fins de comparaisons.

En Côte d'Ivoire dans la collecte de sérum

Un petit questionnaire était posé aux patients qu'il fallait rapprocher à posteriori des diagnostics sur les sérums prélevés.

Une reconstitution écrite des questionnaires, à partir des questionnaires machine avait également été demandée après l'opération aux fins d'archivage et constitution de fiches médicales.

Dans trois enquêtes en vraie grandeur dans la Région Parisienne

L'une pour la CRAMIF en liaison avec le CRESIF pour évaluer auprès de visiteurs l'impact d'une exposition d'éducation pour la santé.

La seconde pour le service de Médecine du Travail de la Société Générale, en liaison également avec le CRESIF pour évaluer à partir d'un échantillon par quotas, l'impact d'une campagne sur les maladies cardio-vasculaires. Cette enquête avait la particularité d'être menée pour moitié sur papier et pour moitié sur micro-ordinateurs de terrain.

La dernière à la Cité des Sciences et de l'Industrie de La Villette auprès des usagers de la médiathèque. Pour cette opération un questionnaire d'une quarantaine de questions a été généré un matin ; la formation des enquêtrices a eu lieu en début d'après-midi ; le questionnaire a ensuite été modifié à la suite de leurs remarques, et l'enquête a commencé le même jour en fin d'après-midi. L'enquête a porté sur 309 questionnaires ; déchargement des données le soir du dernier jour, et sortie des tableaux le lendemain.

D'autres applications sont actuellement en cours d'élaboration ou de négociation, en particulier avec l'appui de l'OMS et de l'UNICEF, d'une part, avec l'appui de l'UNICEF et de la Coopération Française pour la Guinée, d'autre part.

Différents projets SIDA s'intéressent à cette méthodologie, L'INSEE pour ses enquêtes de consommation auprès des ménages. Enfin plusieurs programmes de l'ORSTOM de diverses disciplines envisagent d'utiliser ce progiciel pour leurs opérations de collecte.

8. PRINCIPALES LECONS DES TRAVAUX EFFECTUES, DOMAINES D'APPLICATION

8.1. Principales leçons des travaux effectués

- Faisabilité d'enquêtes sur des machines de terrain et de l'intégration des différentes phases des enquêtes autour de micro-ordinateurs.

- La formation, les coûts, les connexions entre machines ne constituent pas des handicaps importants.

- Nécessité pour l'analyse de partir des tâches du concepteur et de l'enquêteur.

- Aller profondément dans l'analyse pour la correction des erreurs et incohérences pour offrir à l'enquêteur un ensemble d'outils puissants mais simples d'accès.

- L'entrée dans une série de logiciels classiques : dBASE III, SPSS-PC, SAS, ITEM, STATGRAPHICS, etc., ne pose pas de problèmes particuliers.

- Il s'avère que cette approche permet un gain substantiel pour la qualité des données, la rapidité et le coût.

- Il sera facile d'envisager une machine intermédiaire dédiée à la supervision sur le terrain.

- Dans les domaines qui sont les nôtres, notre équipe est considérée comme pionnière tant en France qu'aux Etats-Unis et par divers organismes internationaux.

- Portabilité aisée sur différents types de machines.

8.2. Avantages et contraintes de la saisie informatisée

Contraintes Un certain nombre de contraintes apparaissent a priori :

- il s'agit d'une approche nouvelle, et comme toute nouveauté elle engendre l'inquiétude ;

- l'appréhension vient aussi du fait qu'elle apparaît comme très imbriquée dans des préoccupations informatiques ;

- bien que le prix global d'une opération soit réduit par cette approche, l'impression existe que l'on doit investir beaucoup en équipement. Le chapitre équipement augmente certes, mais d'autres chapitres se trouvent quant à eux réduits très sensiblement ;

- il faut avoir mis au point et testé toute la conception et les contrôles avant de démarrer en vraie grandeur sur le terrain. Mais cette contrainte n'est pas spécifique à cette approche informatisée ; en effet si les opérations classiques étaient bien conçues, la préparation devrait inclure ces mêmes contraintes pour les mener à bien ;

- comme dans toutes les opérations, il faut une méthode et rigueur dans la préparation technique.

Avantages :

- simplicité notamment en supprimant toute la chaîne de gestion des documents ;

- amélioration de la qualité par le contrôle direct à la saisie, et le fait de n'avoir plus d'étape de transcription intermédiaire, source d'erreurs ;

- réduction des délais par suppression du temps de saisie des bordereaux, de retour aux documents de base pour correction éventuelle après contrôle ;

- raccourcissement des délais pour obtenir des résultats déjà pendant l'opération et immédiatement après ;

- réduction des coûts : coût de saisie des bordereaux, d'élaboration d'un programme de contrôle, de correction par retour aux documents de base, moins de déplacements, de personnel, etc. ;

- l'enquêteur est complètement guidé dans sa saisie ; il a moins de questions à se poser ;

- la mise au point des tabulations peut se faire dès avant l'opération et être testée et exécutée dès le retour des premiers enregistrements ;

- cette démarche permet une approche intégrée de l'ensemble d'une opération, depuis la conception jusqu'aux tabulations.

8.3. Domaines d'application et débouchés

Les domaines d'application de ce projet sont vastes et concernent des champs d'activité variés.

Il s'agit principalement de tous les problèmes de saisie sur le terrain qui, variant fréquemment d'objet, sont d'une complexité suffisante pour justifier un logiciel de génération simple de questionnaires et aider à leur passation sur le terrain.

Démographie	Enquêtes quantitatives
Statistique	Suivi d'individus. Fichiers de population
Economie	Collecte d'informations en tous lieux,
Epidémiologie	p.ex. centres d'état-civil, dispensaires
Santé publique	Entrée de mesures cliniques ; Collecte d'informations par des agents (Santé ou collectivités locales, par ex.)
Etudes de marché	
Etudes de consommation et relevé de prix ou de productions	
Enquêtes d'opinion	
Formulaire administratifs sur le terrain	
et toutes saisies sur le terrain	

L'intérêt de ce système ne se limite pas au domaine des enquêtes démographiques ou épidémiologiques ; des applications sont possibles pour tout ce qui concerne les collectes de données par un agent, que ce soit dans un cadre administratif ou d'investigation (formulaire divers, études de marché, sondage, collecte de mesures diverses).

9. L'ETAT DU PROJET ET SA VALORISATION

9.1. Récapitulation des travaux effectués

- Analyse du problème, basée sur un savoir-faire en matière d'enquête et de développement de programmes informatiques.
- Recherche et tests de matériels.
- Financement et appuis.
- Elaboration de l'architecture et des modules.
- Développement du générateur en Turbo-Pascal (Editeur et Compilateur).
- Elaboration d'un langage de description de questionnaires (LDQ).
- Développement en Turbo-Pascal d'un interpréteur pour la machine de terrain.
- Intégration de l'ensemble des programmes en vue d'un prototype.
- Expérimentation-validation par des enquêtes de terrain.
- Recherche de partenaires privés en vue de sa valorisation.

- Etude des développements ultérieurs envisageables :
 - principalement selon la demande ;
 - selon les champs d'application et les types d'enquêtes.
 - selon les types de machine de terrain utilisées ;
 - autres technologies : téléphone, minitel, satellite ;
 - intelligence artificielle, systèmes experts.

9.2. Valorisation

Cette recherche se valorisera principalement par l'association avec le secteur privé pour la commercialisation d'un produit de la recherche publique. Le progiciel SIC est considéré comme pionnier dans la méthodologie des enquêtes.

Autour du progiciel lui-même, d'autres activités valorisantes seront menées telles que la formation, l'assistance technique et les applications dans les domaines santé, démographie, épidémiologie et enquêtes diverses principalement avec les pays en développement.

Enfin ce sont de nouvelles stratégies et méthodologies de collecte qui pourront être développées à partir du SIC.

DISCUSSION SUR LA REALISATION ET L'APPLICATION D'UN LOGICIEL DE TRAITEMENT D'ENQUETES

Agnès Guillaume et Philippe Hamelin
(UR 205, Dynamique des Populations et Culture, Enjeux et
Maîtrise de l'Espace)

Museu Emilio Goeldi
C.P. 075 ORSTOM
66000 Belém, PA (Brésil)

RESUME - La réalisation d'un logiciel de saisie de données semble une tâche relativement simple quand l'on possède quelques connaissances informatiques, une expérience de l'organisation des données et que l'on a identifié les besoins des utilisateurs. Dans ces conditions, en dépassant le budget-temps prévu pour la phase de mise au point, on parvient à diffuser à quelques chercheurs un produit qu'ils utilisent avec satisfaction. Mais l'étape suivante, la transformation du logiciel en un produit de niveau commercial, est trop semé d'embûches pour qu'un chercheur puisse la franchir seul, sans le soutien de services compétents pour aider à la normalisation du produit, en assurer la diffusion et la maintenance.

Nous participions à la réalisation, depuis quelques années, de traitement d'enquêtes à l'aide de logiciels implantés sur le centre de calcul d'Orsay (CIRCE). La maîtrise de l'analyse statistique proprement dite ne posait plus guère de problèmes ; par contre, l'étape saisie et déuration des données ressemblait trop souvent à un casse-tête chinois.

Cette étape exigeait un travail long, souvent rébarbatif et peu créatif. Cette phase du travail avait souvent pour conséquence que l'analyse de certaines enquêtes n'était que très partielle et, dans les cas extrêmes, le seul traitement réalisé était le chargement des données sur bande magnétique et l'archivage de celles-ci.

Ce phénomène était aggravé du fait que certains chercheurs, croyant en la magie du mot ordinateur, avaient un peu perdu de vue les problèmes de la qualité des données et des supports sur lesquels elles étaient recueillies. L'ordinateur était la boîte noire où l'on jetait un paquet de cartes à l'entrée pour récupérer un rapport à la sortie : nous avons enfin trouvé notre pierre philosophale!

Ce mystère s'entretenait d'autant plus facilement qu'avec l'apparition de la statistique descriptive multivariée (analyse des

correspondances, analyses factorielles, etc.), une confusion s'est très vite faite entre la capacité de ces outils à analyser des données hétérogènes (qualitatives, quantitatives, ordinales, etc.) et à analyser n'importe quoi.

Même si le résultat à la sortie de la machine n'était pas probant, l'appel au joker (recodage, lissage, générateur de données manquantes) permettait toujours au bout d'un certain nombre de passages dans la machine d'obtenir un résultat, mais quel résultat ?

Il faut dire que ce système arrangeait tout le monde : le chercheur à qui il évitait une remise en cause de son travail de recueil (de toutes façons les résultats seraient bien plus valorisés par la méthodologie employée que par leur valeur intrinsèque); l'intermédiaire informaticien/statisticien voyait grandir son prestige et devenait un homme indispensable et les institutions de tutelle montraient que leurs troupes faisaient preuve de modernisme.

Nous étions dans la situation paradoxale où l'introduction d'un puissant outil technologique aboutissait à l'apprauvissement des résultats. Cet état de fait se vérifiait surtout dans les sciences où la méthodologie ne possédait pas d'outils mathématiques ou statistiques développés.

L'arrivée en force des micro-ordinateurs, au début des années 80, allait permettre cette démystification de l'ordinateur à travers sa démocratisation. La réalisation d'un logiciel de saisie, contrôle et traitement élémentaire des données nous apparaissait comme une façon de valoriser ce rapprochement de l'ordinateur et du chercheur.

Nous nous sommes cantonnés à la réalisation de traitements statistiques élémentaires, car si la saisie et la vérification de données ne sont que l'automatisation de tâches répétitives ne demandant pas de connaissances théoriques, le fait de savoir positionner cinq paramètres pour lancer une analyse de correspondances ne garantit en rien la maîtrise des présupposés théoriques, qui seuls permettent une interprétation scientifique correcte des tableaux et des graphiques fournis par l'ordinateur. Sans cette connaissance préalable de la statistique, l'interprétation est un acte de divination dans le marc de café.

La deuxième grande préoccupation était le temps : le temps très long entre la passation de l'enquête sur le terrain et la sortie des premiers résultats. Une durée de trois ou quatre ans entre les deux était considérée comme moyenne et les résultats devenaient parfois obsolètes avant publication. Souvent aussi les chercheurs attendaient leur retour en France pour traiter leur enquête, ce qui interdisait le plus souvent tout contrôle *à posteriori* des doutes de toutes sortes qui pouvaient surgir à la vue de certains résultats.

1. LE PROJET

A partir de l'analyse ci-dessus, et à la demande de nos collègues démographes, ont été déterminées les fonctionnalités de ce que devraient être un logiciel informatique capable d'améliorer la qualité des données en rapprochant l'outil informatique du chercheur sur le terrain, de raccourcir cette étape rébarbative entre enquête et analyse, et sous la condition que l'investissement purement informatique de l'utilisateur soit minimum.

1.1. Fonctionnalité générale du logiciel

La facilité d'utilisation, aucune connaissance préalable en informatique ne devrait être requise, ce qui entrainera la réalisation d'un logiciel fermé où l'utilisateur sera guidé pas à pas par une série de menus. Une coupure de courant, où la ballade d'un chat sur le clavier, ne devrait se traduire que par la perte du dernier enregistrement au maximum.

1.2. La structure

Il a été choisi une structure à deux fichiers, un fichier contenant les descripteurs des variables (le dictionnaire) et un fichier contenant les données proprement dites. Cette structure est celle des grands progiciels statistiques (OSIRIS, SPSS, SAS) mis en oeuvre au CIRCE. Les chercheurs en sciences humaines traitant leurs données majoritairement sur ce centre de calcul ne devraient pas, de ce fait, se sentir dépayés.

Une réflexion avait été menée sur la structure hiérarchique des fichiers (économie de place en mémoire de masse), mais cette solution a été abandonnée parce que les difficultés à affronter n'étaient pas à la mesure des résultats espérés. Il faut de toutes façons revenir à une structure rectangulaire pour les analyses : le choix, pour les données très hiérarchisées, a été de les éclater en plusieurs fichiers.

2. LE MATERIEL

Le choix du micro-ordinateur Goupil 3, muni du système d'exploitation Flex 9, était évident à cette époque (décembre 1983), puisque :

- c'était l'unique matériel dont nous disposions ;
- c'était aussi le micro-ordinateur retenu pour la diffusion à l'ORSTOM.

Ce micro-ordinateur était équipé d'un double lecteur de disquettes 8", d'une capacité de un million d'octets. Le choix de

l'interpréteur SBasic allait de soi à cette époque. L'autre choix possible d'un développement en assembleur 6809, malgré la simplicité de cet assembleur, aurait eu pour conséquence une multiplication rédhibitoire du temps de développement.

Le SBasic, par rapport au GWBasic présent sur les machines MS-DOS, présente de nombreux avantages, comme la gestion simple de l'écran et des routines d'erreur ou la présence de macro-fonctions permettant de générer du code (interprétable à l'intérieur du programme lui-même), qui donnait la possibilité de développer facilement des microlangages pour certaines parties du logiciel.

Enfin, la possibilité d'utiliser sous l'interpréteur Basic, à n'importe quel moment, les commandes du système, autorisait la réalisation d'un logiciel totalement fermé.

3. LE LOGICIEL

Il se décompose en trois grands modules et quelques programmes utilitaires. Chaque module est un assemblage de petits programmes (au maximum 250 lignes), qui sont chaînés les uns aux autres à travers une série de menus. Cela était nécessaire car nous ne disposions, après le chargement de l'interpréteur Basic, de guère plus de 30K, pour les programmes et les données.

Les fichiers sont du type à accès direct. Les données numériques sont enregistrées en binaire sur deux octets et les données alphabétiques en caractères. En réalité les données numériques et alphabétiques sont situées sur deux fichiers parallèles, mais cela reste totalement transparent pour l'utilisateur.

3.1. Le module dictionnaire

Le module dictionnaire est composé de quatre programmes :

- le premier assure la création du fichier dictionnaire et du fichier de données, ainsi que leur protection contre toute destruction intempestive ;

- le deuxième enregistre, à travers une saisie contrôlée, les descripteurs de chaque variable. Ces descripteurs de variables sont au nombre de huit : le nom, le nom abrégé, le type, la longueur, la valeur minimale, la valeur maximale, le nombre de valeurs d'exclusions (valeurs en dehors des bornes extrêmes qui servent à repérer, par exemple pour les âges, les gens dont on ne connaît pas l'âge, mais dont on sait s'ils sont adultes ou enfants), et enfin la valeur de la "sans réponse" ;

- le troisième permettra la visualisation, l'impression et éventuellement la correction des données enregistrées avec le deuxième programme ;

- le quatrième assurera, à l'aide de données enregistrées avec le deuxième programme, la génération des paramètres du masque de saisie. Il calculera, en fonction du nombre d'individus prévus en saisie, la taille à réserver sur la disquette et effectuera cette réservation. En cas de fichier dépassant la taille d'une disquette il fera les réservations nécessaires (une procédure est prévue au niveau de la saisie pour pouvoir augmenter la capacité disponible).

3.2. Le module saisie

Trois programmes composent le module de saisie des données :

- le programme de saisie qui assure l'affichage de la grille à l'écran, la gestion du curseur, le contrôle de la longueur de la variable et si celle-ci se situe bien entre les bornes définies ou correspond à une valeur d'exception. La touche "*retour chariot*" correspond au code sans réponse quand une valeur a été définie lors de la création du dictionnaire : dans le cas où aucune valeur n'a été indiquée, l'opérateur doit entrer obligatoirement une valeur. Cela est pratique : dans le cas où une partie des réponses est facultative, il est très conseillé alors de mettre une valeur obligatoire en en-tête de chaque nouveau paragraphe ;

- un programme permettant l'impression des enregistrements sélectionnés par les valeurs d'une borne inférieure et d'une borne supérieure, soit à l'écran, soit sur l'imprimante ;

- un programme de correction/vérification des données qui permet, soit de visualiser séquentiellement chaque enregistrement en avant ou en arrière, soit de se déplacer aléatoirement en fournissant un numéro d'enregistrement. Les valeurs entrées lors de la correction sont, bien entendu, vérifiées de la même façon que lors de la saisie. En réalité ce programme est bien plus performant que le programme de saisie.

Ce module pourrait être simplifié en fusionnant le programme saisie et le programme correction en un seul fichier, ce qui serait relativement simple et il suffirait de quelques ajouts mineurs au programme de correction pour qu'il puisse aussi assurer la saisie.

3.3. Le module traitement

Comme il a été dit par ailleurs, la vocation de ce module n'était pas le traitement scientifique des données mais plutôt un ensemble permettant la sortie rapide des résultats et de vérifier ainsi la cohérence, aussi bien interne qu'externe, des données.

Si la cohérence interne ne demande généralement qu'un retour au questionnaire pour être validée, la cohérence externe, mise en évidence par des tabulations simples ou croisées, demande

bien souvent un retour sur le terrain pour être décidée, d'où l'importance d'une disponibilité rapide des résultats.

Le programme de contrôle de cohérence est prévu pour tester la cohérence des réponses à deux ou plusieurs variables. Pour formuler les équations à tester, l'opérateur a à sa disposition un micro-langage qui contient comme alphabet soit des constantes numériques, soit les variables numériques décrites dans le dictionnaire, des opérateurs logiques (*ET*, *OU*) et des opérateurs de relation (=, <, >, <=, >=, <>). Cela est complété par le choix laissé à l'opérateur d'extraire la sous-population définie par l'équation ou son complément.

Deux modes de fonctionnement sont proposés :

- soit *interactif*. L'opérateur rentre l'équation et obtient le résultat de suite à l'écran ;
- soit *en temps différé*. L'opérateur rentre une série d'équations et lance l'exécution pratique lorsque les fichiers sont importants.

L'utilisation de ce programme permet l'extraction de sous-populations.

Le programme de tris à plat est tout simple : il suffit de fournir comme paramètre l'intervalle des variables que l'on désire traiter.

Le programme de tabulations croisées permet de traiter simultanément une, deux ou trois variables. Il existe une possibilité de filtrer les individus statistiques. La construction des filtres à l'aide du même micro-langage que pour les contrôles de cohérence. Il est aussi possible de redéfinir le minimum, le maximum et les valeurs d'exception de chacune des variables : dans ce cas les valeurs apparaissant en dehors des nouvelles limites rejeteront l'individu dans la catégorie rebut, qui ne sera pas prise en compte lors du calcul des statistiques.

Enfin, un petit programme permet de transformer les fichiers de données en binaire et à accès direct, en fichiers séquentiels EBCDIC, en vue de leur transfert et/ou de leur utilisation avec d'autres progiciels de traitements statistiques.

En plus des utilitaires de sauvegarde et de formatage de disquettes, il a été réalisé au Togo, par l'équipe Levy/Pilon, d'autres programmes, en particulier, un de recodage qui faisait défaut.

4. LA REALISATION

Le projet était peu ambitieux au départ, la durée prévue pour sa réalisation était de deux mois. Il est vrai que fin janvier une première version du logiciel était mise en test, le module traitement

ne comportant alors qu'un programme. Mais au minimum deux mois et demi ont été nécessaires pour les tests et la mise au point.

Nous avons sous-estimé la durée de la phase test et mise au point, qui est certainement la plus longue. Nous estimons la phase définition du projet aux seuls quinze jours de décembre pendant lesquels nous avons défini les fonctionnalités du projet et à un mois l'écriture des programmes.

L'impasse faite sur l'analyse organique a eu un impact direct sur une écriture embrouillée des programmes, qui sont parfois d'une maintenance difficile.

Il manque aussi une documentation des programmes ; par contre le manuel de l'utilisateur a été rédigé.

Nous nous sommes aperçus alors que le coût de développement d'un tel logiciel est très élevé, l'estimation minimum est de 200 000 francs (main d'oeuvre, matériel, locaux), ce coût aurait certainement doublé pour mettre le produit à une norme commerciale. heureusement qu'il n'existait pas de produit semblable disponible sur le marché!

5. LES APPLICATIONS

L'enquête mortalité prénatale en Guadeloupe (enquête conjointe ORSTOM, DASS - Guadeloupe et INSERM) a été la première application de ce logiciel. La demande avait été formulée par J.P. Guengant (UR 709), qui souhaitait pouvoir exploiter cette enquête sur place afin d'en contrôler les résultats au fur et à mesure de sa réalisation.

L'enquête avait pour originalités :

- sa durée dans le temps, deux ans ;
- sa longueur et la complexité du questionnaire, 343

variables.

L'utilisation du logiciel s'est faite sans problème par des personnes sans aucune formation informatique. Une secrétaire de la DASS, qui n'avait encore jamais pratiqué l'informatique, a assuré la saisie et la correction des données. Un des médecins enquêteurs a présenté des tableaux de résultats de l'année 1984 lors d'un séminaire en février 1985. Il faut souligner que ce médecin utilisait pour la première fois l'informatique.

Une version disque dur du logiciel fût réalisée fin 1984 pour le traitement d'une enquête pluridisciplinaire au Nord-Togo. Une panne de disque dur seulement résolue à Paris fût le problème le plus important affronté par cette équipe.

Cette enquête se décomposait en sept fichiers, qui représentaient environ un total de quatre millions d'octets à saisir. Les chercheurs ont apprécié la facilité du micro-langage pour les demandes

de contrôle de cohérence, mais ont regretté sa lenteur d'exécution sur gros fichiers.

Par ailleurs ils ont fait développer des compléments par des informaticiens locaux. Ces modules complémentaires ont bien fonctionné, ce qui pourrait signifier que notre programmation n'était pas aussi illisible que nous le croyions.

Au Togo, il a été à nouveau utilisé pour une enquête épidémiologique, en Côte d'Ivoire pour la saisie des registres de dispensaire et aussi pour d'autres saisies en Guadeloupe.

6. LE BILAN

D'un point de vue strictement comptable, nous pouvons dire que le projet a été au moins amorti par les utilisateurs, en terme de gain de temps et d'une meilleure qualité des données. Mais si l'on considère que ce projet était le ré-investissement de plusieurs années d'expériences acquises par les auteurs, c'est un échec du point de vue de la valorisation de notre savoir-faire.

Les causes sont principalement de deux ordres :

- l'arrivée sur le marché et son hégémonie rapide du standard IBM-PC, qui rendait notre logiciel incompatible avec une grande partie du nouveau parc ordinateur de l'ORSTOM. Si ce logiciel avait pu être rendu compatible IBM-PC par une conversion sous MS-DOS (travail non réalisé par manque de temps de notre part, mais aussi par manque de soutien), son utilisation aurait été assurée pour des travaux en Côte d'Ivoire, Indonésie, Brésil et Mexique ;

- le manque d'infrastructure à l'intérieur de l'ORSTOM pour maintenir et diffuser les logiciels. Les deux auteurs avaient entrepris une recherche méthodologique finalisée : il n'entraît ni dans leur compétence, ni dans leur intention de se transformer en commerciaux pour vendre leur produit. Nous nous étions adressés à l'Unité de Valorisation de la DIVA pour examiner conjointement les suites qui pourraient être données et de quelle manière, mais nos démarches n'ont pas abouti. Comme d'autres engagements allaient bientôt nous accaparer, nous décidions alors d'arrêter là notre projet : nous imaginions mal comment les deux auteurs de ce projet, l'un étant au Brésil, l'autre en Côte d'Ivoire, pourraient assurer la conversion, la diffusion et la maintenance de ce logiciel.

Si le premier point est inhérent à la dynamique du développement de l'informatique, le deuxième doit être résolu avant d'envisager le développement de tout logiciel d'intérêt général.

La création d'une structure à l'ORSTOM capable, d'une part, d'évaluer les projets soumis pour décider de l'opportunité de les développer et, d'autre part, fournir une assistance technique

durant la phase de développement ainsi que d'en assurer par la suite la diffusion et le suivi est très souhaitable.

La puissance des micro-ordinateurs et la diffusion des disques durs permettent maintenant l'utilisation de puissants logiciels de traitement de données, qui, jusqu'alors, étaient réservés aux centres serveurs : se résout ainsi le problème "*analyse des données*", néanmoins le problème de l'acquisition des données demeure.

Avant de développer un ou plusieurs logiciels de saisie, il semble indispensable d'entamer une réflexion approfondie sur les différents types de données qui peuvent se présenter. Nous pouvons déjà en distinguer deux :

- les données fournies par les instruments de mesure, où le problème est l'automatisation complète de la chaîne ;
- celles recueillies par un observateur humain, subdivisées en deux groupes :
 - les enquêtes lourdes qui recueillent une grande masse de données, dans un temps limité ;
 - les panels, les enquêtes à passages répétés, les observatoires, etc., soit toutes les structures qui recueillent des informations étalées dans le temps et dont le volume moyen journalier est raisonnable.

En général, la disponibilité de l'information déjà recueillie peut être d'un grand bénéfice. Si dans le premier cas le recours à des services spécialisés semble la meilleure solution, pour des raisons d'efficacité (durée de saisie, personnel temporaire qualifié), l'introduction du micro-ordinateur dans le second cas devrait dépasser le simple stade du convertisseur, pour fournir : suivi de planning, fiche enquêteur, bordereaux du prochain passage, etc..

Le futur est peut-être le micro-ordinateur portable, qui substituera questionnaire et cahier de notes. Une réflexion doit s'engager sur cette perspective qui bousculera beaucoup de concepts en matière d'acquisition de données.

CONCLUSION

L'ORSTOM n'a ni la vocation, ni les moyens, de développer une recherche en informatique, mais doit rechercher la meilleure utilisation possible de cet outil et nous croyons qu'à l'avenir deux grandes voies, qu'il ne faut impérativement pas confondre, se dessinent :

- *le service chercheur*. L'ordinateur comme bonne à tout faire du chercheur : le chercheur s'informatise pour gagner du temps lors des tâches de routine (traitement de texte, gestion de bibliographie, consultation de bases de données, calculs, etc.) ;

- *la valorisation*. Des recherches bien menées, qui aboutissent à des méthodologies fiables ou, aussi, des méthodologies développées parallèlement à certaines recherches d'un intérêt général peuvent être valorisées à travers la réalisation d'un produit informatique.

Cette transmission, à travers de l'informatique, du *savoir-faire* du chercheur équivaut, peut-être avec un sens plus large, à la réalisation du système-expert décrit par Pascal Renaud (Cf. ORSTOM Actualités, n°17). Un système-expert à l'intention des décideurs et des spécialistes chargés de l'application, mais aussi à l'intention des chercheurs d'autres spécialités confrontés à des problèmes spécifiques.

Encore faut-il bien faire la distinction entre expert et chercheur, le premier se caractérise par un diagnostic rapide grâce au savoir-faire accumulé, tandis que le second recherche l'accumulation de nouvelles connaissances à travers l'analyse d'un phénomène-recherche. Ce sont deux métiers différents, même si le chercheur joue le rôle d'expert, de temps à autre.

Cela permettrait peut-être de clarifier les débats sur la recherche pluridisciplinaire, car bien des projets dits "*pluridisciplinaires*" ne sont bien souvent que des projets monodisciplinaires qui nécessitent l'intervention de consultants ou d'experts de divers autres champs de connaissance.

L'emploi de tels systèmes ne doit, en aucun cas, tuer la créativité du chercheur et il faut rappeler que l'informatique est l'outil et que jamais une enquête ou un recueil de données ne peut-être considéré mauvais, parce que non traitable par tel ou tel logiciel informatique : ce serait au contraire la preuve de l'inadéquation, ou de la faiblesse, de ce logiciel.

TELEOBSERVATION INFORMATISEE DES VOLCANS ACTIFS

Marc Henrotte, Michel Lardy et Jeanne Tabbagh
(Géophysique)
(CRG, GARCHY et UR 106, NOUMEA)

avec la collaboration de :
Christian ARCHAMBAULT, Claude PAMBRUN
Ingénieurs CNET et CNRS

et
Jean-Philippe EISSEN, Patrick MAILLET, Michel MONZIER
Pétrographes

RESUME - L'ORSTOM, dans le cadre du Programme Interdisciplinaire de Recherche sur la Prévision et la Surveillance des Eruptions Volcaniques (PIRPSEV), assure une surveillance ininterrompue du volcan Matthew (S.O. Pacifique) depuis octobre 1986, en utilisant un équipement automatique qui transfère un ensemble de quinze paramètres géophysiques et météorologiques, via le système satellitaire ARGOS, au centre de distribution de données de Toulouse. Ces données sont accessibles au centre de calcul du CIRCE à Orsay, après une remise en forme par le Centre de Recherches Géophysiques (CRG).

Le Centre de Nouméa, éloigné d'environ 500 km du volcan, récupère les données par l'intermédiaire du réseau TOMPAC (TRANSPAC), à l'aide d'un micro-ordinateur PC-compatible et d'un modem connecté sur le réseau téléphonique commuté. Les données d'un passage satellite sont prélevées chaque jour, afin d'assurer une surveillance régulière de l'édifice volcanique.

Le volcanisme témoigne de la dynamique de notre planète, il en est une des manifestations les plus spectaculaires ; que l'activité volcanique soit associée à la création (dorsale) ou à la destruction (arcs insulaires, marges actives) de croûte océanique, ou encore qu'elle se manifeste sous forme de "*points chauds*" (intraplaque) l'étude de ces phénomènes reste essentielle pour la compréhension de la vie de la terre.

1. SURVEILLANCE PERMANENTE DES VOLCANS (*)

Les mesures faites par des groupes de spécialistes de diverses disciplines, autour d'observatoires permanents disposés sur quelques volcans nécessitent des moyens importants qu'il paraît difficile de mettre systématiquement en oeuvre sur les quelques 500 édifices actifs à la surface de la terre, tant le déploiement de tels observatoires est coûteux. Ces opérations qui restent essentielles pour une meilleure compréhension des mécanismes volcaniques et des phénomènes associés à la dynamique de notre planète peuvent être complétées par des observatoires plus simples.

Les techniques spatiales permettent d'assurer des mesures automatiques et permanentes pour un ensemble de paramètres significatifs en regard des édifices concernés. Un suivi dans le temps de l'évolution des volcans peut ainsi contribuer au *pilotage d'opérations lourdes et approfondies, mais limitées dans le temps.*

2. MOYENS D'INVESTIGATIONS

La téléobservation des volcans actifs par l'utilisation des télétransmissions spatiales (ARGOS) pour l'acquisition régulière de données, ou l'utilisation des techniques de télédétection pour des études de thermographie ou géomorphologie sont des *"moyens d'approfondissement et d'élargissement des investigations nécessaires en volcanologie"*.

Les techniques spatiales sont généralement associées à de gros moyens informatiques et télématiques qui facilitent le traitement et la distribution de l'information recueillie auprès des pays dont les volcans sont sous surveillance et de la communauté scientifique concernée.

Deux modèles d'unité d'acquisition de terrain (**), associés au système ARGOS ont été développés en France ces dernières années. L'expérience acquise par le CNET et le PIRPSEV sur le *"volcan laboratoire"* Etna a été transposée et adaptée au volcan Matthew (Pacifique Sud) de type explosif et aux éruptions espacées.

3. MATTHEW

Un avenant à l'accord cadre ORSTOM-CNRS a permis d'officialiser la collaboration entre le CNET, l'ORSTOM et le PIRPSEV pour la mise en place d'un observatoire permanent sur

un volcan actif de l'extrémité méridionale de l'arc insulaire des Nouvelles Hébrides.

L'installation effectuée pendant le mois de septembre 1986 aura permis de récupérer près de 50000 données au 1er octobre 1987, qui seront volontairement réduites à deux fichiers d'environ 33000 valeurs brutes et interprétées. Les deux fichiers sont en permanence accessibles au CIRCE par l'ensemble des moyens télématiques actuels.

4. CHEMINEMENT DES DONNÉES

Les quinze paramètres numérisés émis par la balise ARGOS toutes les deux minutes et demi sont recueillis par un des deux satellites TIROS de façon héliosynchrone toutes les deux à trois heures. Après l'émission des données vers une des trois stations terrestres de réception, les messages deviennent accessibles par les différents utilisateurs auprès d'un des deux centres de distribution (Centres CLS de Toulouse et Washington).

Les données acquises sur Matthew et actualisées toutes les quatre heures portent sur la mesure des températures de fumeroles, de températures de sol le long d'un profil vertical et de comptage de dépassements de seuils par un signal sismologique ; elles sont également complétées par des mesures de météorologie.

Le centre ORSTOM de Nouméa assure une surveillance quotidienne par interrogation du centre de Toulouse via le réseau TOMPAC (TRANSPAC-TOM) mis en place en juillet 1986 par la DGT et l'OPT de N-C.

La centralisation, le traitement, et l'archivage des informations pour la constitution d'une base de données accessible dans le plus court délai sont assurés par le CRG de Garchy. L'acquisition des données auprès du centre serveur ARGOS de Toulouse à travers le réseau TRANSPAC est assuré automatiquement deux fois par jour par un microordinateur du type HP 85 ; un changement de format assure le compactage des fichiers qui sont stockés sur cassette. Ces données sont ensuite envoyées automatiquement au CIRCE où elles sont traitées. Un ensemble de logiciels permet de trier ces données suivant l'heure de réception des messages, d'éliminer les doubles et de constituer un fichier de données brutes par mois qui est consultable immédiatement. Un traitement plus complet est effectué une fois par jour avant huit heures : ce traitement consiste à éliminer les erreurs de transmission, à calculer l'heure d'acquisition et la valeur du paramètre physique à partir des formules de conversion et des constantes d'étalonnage, afin de créer un fichier de données interprétées par mois. Ces opérations s'effectuant toujours sous le contrôle du micro-ordinateur. A tout

moment chaque membre de la communauté scientifique peut accéder à ces données, le CIRCE pouvant être appelé par l'ensemble des moyens télématiques actuels : console passive, terminal personnel, réseaux EARN et Minitel. Un ensemble de logiciels, accessible à des non-informaticiens, a été mis au point permettant à chacun d'obtenir sur un Minitel, chaque matin, sous forme de courbes la variation quotidienne ou mensuelle d'un paramètre ainsi que les valeurs numériques des paramètres physiques interrogés (exemples page suivante).

Grâce à ces procédures, l'accès le plus large et le plus rapide possible est offert à une communauté d'utilisateurs ; c'est ainsi que les données interprétées disponibles au CIRCE sont prélevées une fois par mois depuis le centre de Nouméa par le réseau TOMPAC pour être stockées sur une disquette compatible afin de permettre un traitement ultérieur. L'utilisation de logiciels du commerce adaptés permettent de traiter les divers paramètres à la demande de l'utilisateur sans qu'une grande connaissance informatique soit nécessaire.

Pour faire face à l'augmentation prévue du nombre de balises et diminuer les coûts de calcul, l'ensemble de la chaîne de traitement sera prochainement effectué à l'aide d'un micro-ordinateur type IBM AT, le CIRCE restant le coeur de l'archivage et de l'interrogation extérieure.

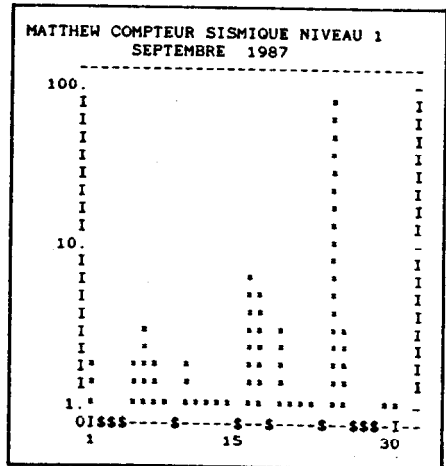
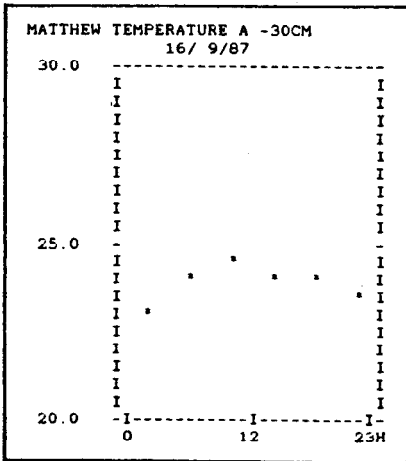
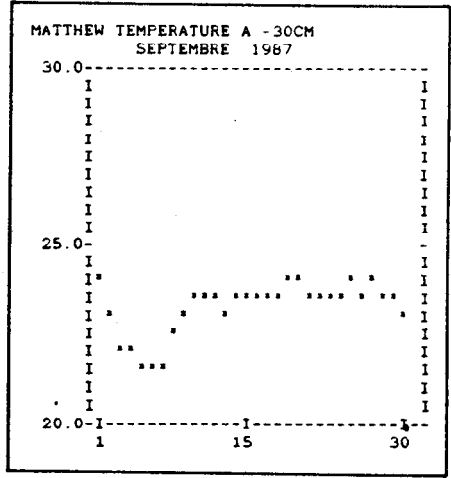
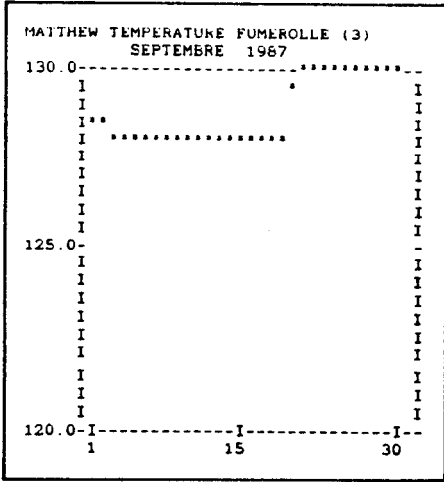
CONCLUSION

A partir de données caractéristiques et grâce à l'utilisation de moyens télématiques, il est possible de suivre en temps quasi réel l'évolution d'un édifice volcanique. Cela permet à l'ensemble de la communauté scientifique de disposer de messages clairs, faciles à obtenir à partir d'un simple poste de téléphone par l'intermédiaire du réseau commuté ou d'un réseau spécialisé.

(*) : Bull. ARGOS Newsletter n 30. Téléobservation des volcans actifs : enjeux, difficultés, perspectives. juillet 1987. Remote monitoring of active volcanals : challenge, problems, outlook.

(**) : Centre de recherches volcanologiques du CNRS
 LGIT Grenoble
 CNET Lannion et PIRPSEV (INSU Paris)

Exemples de données graphiques quotidiennes
ou mensuelles obtenues sur Minitel



```

      C. T. I. V.
    VOLCAN M A T T H E W
PARAMETRES DISPONIBLES
NUMERO DE MESSAGE-----(1)
TEMPERATURE SOUS ABRI-----(2)
PROFIL VERTICAL DE TEMPERATURE
      120 CM-----(3)
      60 CM-----(4)
      30 CM-----(5)
      3 CM-----(6)
TENSION DE BATTERIE-----(7)
PRESSION ATMOSPHERIQUE-----(8)
HUMIDITE RELATIVE -----(9)
TEMPERATURE DE FUMEROLLES
      CAPTEUR 1-----(10)
      CAPTEUR 2-----(11)
      CAPTEUR 3-----(12)
COMPTEURS D'EVENEMENTS SISMIQUES
      NIVEAU 1-----(13)
      NIVEAU 2-----(14)
      NIVEAU 3-----(15)
NO DU PARAMETRE CHOISI :5
VOULEZ-VOUS LE TRACE DU MOIS (1)?
      OU DE QUELQUES JOURS (2)?

```

Menu obtenu sur MINITEL après envoi du mot de passe

```

1 ER JOUR DU MOIS DESIRE ? (1-31)
?
17
DERNIER JOUR DU MOIS DESIRE ?
?
22
17/11/87 2H42      70 23.4
17/11/87 6H42      70 23.5
17/11/87 10H41     69 23.3
17/11/87 14H40     69 23.2
17/11/87 18H40     70 23.4
18/11/87 2H39      70 23.4
18/11/87 6H38      70 23.5
18/11/87 10H38     70 23.2
18/11/87 14H37     70 23.2
18/11/87 18H37     70 23.6
18/11/87 22H38     70 23.9
19/11/87 2H39      70 23.4
19/11/87 6H40      71 23.4
19/11/87 10H41     64 23.3
19/11/87 14H42     67 23.2
19/11/87 18H43     70 23.2
19/11/87 22H44     70 23.8
20/11/87 2H45      71 23.5
20/11/87 6H47      71 23.3
20/11/87 10H49     70 23.2
20/11/87 14H51     70 23.1
20/11/87 18H53     70 24.4
20/11/87 22H55     70 24.1

```

Liste des valeurs d'une température de fumerolle et de sol en degrés C, obtenues à chaque passage satellite entre le 17 et le 20 novembre

GLOSSAIRE

CIRCE : Centre inter-régional de calcul électronique (CNRS)

CLS : Collecte et localisation de données satellitaires (ARGOS)

CNET : Centre National d'études des télécommunications

CRG : Centre de recherches géophysiques, Garchy (CNRS)

DGT : Direction générale des télécommunications

EARN : Réseau IBM

OPT : Office des postes et télécommunications de Nouvelle Calédonie

PIRPSEV : Programme interdisciplinaire de recherche sur la prévision et la surveillance des éruptions volcaniques

TOMPAC : Equivalent Transpac pour les territoires d'outre-mer

TRANSPAC : Transmission de données par paquet.

INTERET, CREATION ET UTILISATION DE GRILLES D'ECRAN

Patrick Séchet
(Informatique)

(UR 502, Cadres Spatiaux de l'Indépendance Alimentaire)

EMBRAPA

Empresa Brasileira de Pesquisa Agropecuária
SuperCenter Venancio 2.000
70333 - Brasília, DF (Brésil)

RESUME - En matière de langage, on a le choix entre deux possibilités pour la mise au point d'une application sur micro-ordinateur : adopter un progiciel spécifique de gestion de données ou utiliser un langage de programmation.

La première option est très séduisante, surtout parce qu'elle permet de disposer immédiatement de diverses ressources comme la saisie formatée, la génération d'états, les menus d'aide, etc.. Toutefois, on finit toujours par se heurter à telle ou telle restriction inhérente à la méthode d'accès, à la structure des données, aux possibilités de calculs mathématiques, à la capacité de génération de graphiques, à l'utilisation de tables de codification-décodification, etc..

D'un autre côté, en utilisant un langage de programmation on souffre de l'absence de ressources satisfaisantes pour l'entrée des informations. Cette note présente la réalisation d'un éditeur et gestionnaire de grilles d'écran qui favorise le dialogue entre l'utilisateur et le système, en facilitant la saisie des données.

Les versions les plus récentes, opérationnelles sur micro-ordinateurs CP/M 80 et MS-DOS, ont été développées en langage Pascal par l'équipe franco-brésilienne du programme SISGEO : les commandes de manipulation du curseur, édition de texte et saisie sont conformes au standard introduit par le progiciel de traitement de texte *"Wordstar"*.

Cet outil est constitué de deux éléments distincts : un éditeur pour dessiner les grilles d'écran et un sous-programme d'accès à ces écrans, le gestionnaire d'écrans.

L'éditeur de grilles d'écran est un programme qui produit et manipule des fichiers normalisés qui contiennent l'image et la définition des grilles qui seront utilisées par les programmes applicatifs. Durant une session d'édition, on a toutes les facilités d'un

traitement de textes pour dessiner la grille sur l'écran, telle qu'elle est souhaitée par le développeur.

Chaque caractère de la grille est soit une constante, soit une variable, non connue au moment du dessin de l'écran et que l'on représente par un caractère spécial. On distingue dans ce dernier groupe les champs accessibles par l'utilisateur (options, données captées) des champs protégés, non accessibles.

Ces derniers peuvent être élaborés à partir d'informations disponibles ou simplement obtenus par l'intermédiaire d'une grille précédente. Ils sont restitués, du système à l'utilisateur, par l'intermédiaire de la grille d'écran (date de session, par exemple).

Un ensemble d'options de l'éditeur fournit les fonctions nécessaires à la manipulation de ces fichiers. En particulier, l'édition des grilles est permise sous deux formes différentes :

- sous forme de grille d'écran simple (image de ce qui sera présenté sur le moniteur), utilisée pour le dialogue avec le commanditaire du travail et dans la documentation du système ;

- sous forme de grille d'écran accompagnée d'un dictionnaire des variables correspondantes (numéro, position sur l'écran et format d'édition de chaque champ variable), utilisé par le programmeur pour la codification.

Le gestionnaire de grilles d'écran est une subroutine appelée par les programmes applicatifs et sert d'interface avec les écrans. Il fournit essentiellement trois fonctions :

- affichage d'une grille d'écran, ce qui suppose l'ouverture et la lecture d'un fichier correspondant (ou la recherche dans une bibliothèque d'écrans), et la visualisation proprement dite ;

- écriture d'une valeur dans le champ variable correspondant, protégé ou non, destiné à informer l'utilisateur du système ;

- récupération de l'information fournie par l'opérateur à partir de la grille d'écran et mise à disposition du programme hôte. L'utilisateur dispose pour la saisie de toutes les capacités d'édition "*plein écran*". Trois types de champs sont définis : numérique fixe, numérique "*flottant*" et alphanumérique.

Le transfert des informations saisies avec le programme hôte se fait à chaque champ obtenu, ce qui apporte un maximum de souplesse pour la validation, qui doit être codifiée dans le programme principal.

L'adoption d'un outil de ce type est de nature à optimiser la mise au point des systèmes applicatifs qui font intervenir de la saisie de données sur écrans formatés. Il favorise en outre la normalisation des programmes produits (donc la maintenance ultérieure) et standardise également l'interface avec les utilisateurs de ces

programmes. Enfin, un avantage (et non des moindres) est qu'il génère sa propre part de documentation.

1. JUSTIFICATION

L'aspect conversationnel de la relation entre l'homme et la machine est, sans aucun doute, l'un des avantages prépondérants que la micro-informatique a apporté par rapport à l'informatique traditionnelle. Cet aspect est particulièrement sensible à la simplicité syntaxique du dialogue avec l'ordinateur et à la brièveté du temps de réponse de celui-ci, qualités que l'on ne rencontre pas toujours lors de l'utilisation de ressources centrales partagées.

Ce dialogue s'établit, par l'intermédiaire du clavier et de l'écran, et se compose d'une succession d'informations, tantôt fournies par l'ordinateur, tantôt par l'opérateur.

En utilisant les ressources de base des langages de programmation, un échange de données en temps d'exécution pourra être effectué, une ligne après l'autre, par une séquence d'ordres de lecture et d'écriture à l'écran (dans la plupart des langages : Read pour une information à fournir par l'opérateur, Write pour une réponse donnée par le système).

Le dialogue créé de cette façon est assez précaire. En effet, lorsque l'écran est plein, l'affichage d'une nouvelle ligne provoque un déplacement vers le haut ("*scroll-up*") de l'ensemble des informations présentes sur l'écran, la ligne située au sommet disparaissant. Il en résulte une perception séquentielle et partielle, la plupart du temps non satisfaisante, des informations échangées. De plus, les commandes d'édition de champs dont on dispose pour recevoir des informations de la part de l'utilisateur du système sont rudimentaires : le plus souvent seule la touche ("*backspace*") de retour arrière avec effacement est opérationnelle pour une instruction read, permettant la correction d'une entrée erronée.

Pour améliorer ce système de communication, on a d'abord imaginé un dialogue par le biais d'écrans pleins et, plus récemment, de portions d'écran (fenêtres), en considérant un écran comme un module capable de véhiculer les informations, dans les deux sens, de forme tout à fait semblable à un formulaire. Les écrans servent alors à présenter des informations à un utilisateur, à lui poser des questions, lui présenter une suite d'options, fournir une grille de saisie, etc..

Ceci constitue un moyen efficace de structurer le dialogue, dans la mesure où l'on projette chaque fenêtre pour qu'elle contienne des données appartenant à un même groupe, exactement comme l'on dessine un gabarit de rapport ou le "*lay-out*" d'un bordereau.

Ce type de dialogue structuré est adopté par un grand nombre de progiciels de toute nature : il est généralement facile à réaliser avec les langages de prototypage, ce qui est une des raisons pour lesquelles beaucoup de systèmes sont mis au point avec ces derniers.

Parmi les applications qui tirent le plus grand bénéfice de cette manière de travailler on peut distinguer tous les systèmes destinés à la capitalisation d'information, qui utilisent les écrans pour la saisie des données : en particulier les systèmes destinés au dépouillement et traitement des enquêtes. Bien entendu, la réalisation du dialogue système-utilisateur par le biais de fenêtres peut être introduit pour les consultations structurées, la mise à jour d'informations, l'aide en ligne, etc., jusqu'aux simples écrans de menus, de telle façon qu'une application simple sur micro-ordinateur pourra devoir administrer plusieurs fenêtres, allant souvent jusqu'à quelques dizaines.

2. AVANTAGES

Il est bien sûr tout à fait possible de mettre au point, de forme indépendante dans chaque programme d'application, les ressources de visualisation et récupération d'information par le biais de fenêtres. Toutefois, l'utilisation d'un module spécifique d'administration de ces écrans apporte un certain nombre de bénéfices et permet de satisfaire à plusieurs objectifs dans le contexte de développement de progiciels.

Telle méthode permet en effet :

- *d'accélérer la mise au point.* Une fois que le programmeur domine cet outil, toutes les fonctions d'entrée-sortie par écran des programmes sont facilement et rapidement codifiées. Il est très difficile d'évaluer le gain de temps qui peut être apporté durant cette phase, compte tenu de la diversité des applications possibles. On peut néanmoins signaler qu'il existe toujours une interface de communication entre le progiciel et son utilisateur (par conséquent toutes les applications sont concernées). Celle-ci constitue parfois l'essentiel du système, par exemple lorsque le but poursuivi est de stocker des données : dans ce dernier cas l'effort de programmation pourra être réduit substantiellement par l'utilisation d'un gestionnaire d'écrans ;

- *de réduire la taille des programmes.* Fenêtres et écrans sont en effet traités comme des fichiers externes aux programmes, de sorte que la taille de ces derniers est réduite, aussi bien en ce qui concerne le code source que le module objet. Ces fichiers peuvent d'ailleurs être partagés par plusieurs programmes (écrans d'aide, par exemple), ce qui constitue aussi un facteur de réduction de la

taille des programmes, tout en évitant des redondances. De plus, le traitement d'un écran est normalement réalisé avec quelques lignes de code (appels de procédures ou fonctions), en remplacement de toute une succession d'ordres de lecture et d'écriture de la méthode conventionnelle. Enfin, un gestionnaire d'écrans pourra parfaitement se charger de l'essentiel de la validation des informations fournies par l'utilisateur, dispensant le programme applicatif d'encombrantes routines de validation et manipulation d'erreurs. L'avantage de réduction de la taille des programmes est particulièrement significatif lorsque la mémoire interne est limitée : dans tous les cas, le travail de programmation est réduit, non seulement parce qu'il y a moins de code, mais aussi parce que celui-ci est mieux structuré et plus simple ;

- *de promouvoir l'indépendance* entre le dialogue et les programmes. Toute modification de la communication entre le progiciel et son utilisateur se traduira par une mise à jour de l'écran correspondant, sans nécessairement avoir recours à une modification de programme. Alors que dans un environnement traditionnel il est nécessaire de rechercher et modifier plusieurs ordres de lecture et d'écriture dans un programme au moins ; si l'on utilise un gestionnaire d'écrans, il suffira d'éditer l'écran à modifier pour réaliser les changements voulus ;

- *de normaliser les programmes*. La communication entre les programmes et le moniteur utilise un unique intermédiaire, le gestionnaire d'écrans. Celui-ci étant le même pour tous les programmes, même développés par des personnes différentes, dans le cadre d'un même système applicatif, il s'ensuit que les tests et la maintenance s'en trouvent grandement facilités ;

- *de standardiser l'interface avec l'utilisateur*. Ce dernier, en communiquant avec le progiciel, utilise normalement plusieurs programmes, sans forcément le savoir. Il lui est alors difficilement compréhensible que l'appui d'une même touche puisse avoir des effets différents, lors de phases distinctes du traitement. L'utilisation du gestionnaire (Figure 1) est une garantie de ce que l'utilisateur peut attendre un comportement unique de toutes les ressources du progiciel, en ce qui concerne la fonction des touches, les facilités d'entrée de données, la présentation des messages, etc. ;

- *de favoriser la maintenabilité*. Les ordres de lecture et écriture, ajoutés aux routines de validation, occupent une partie significative de beaucoup de programmes : le fait que l'ensemble puisse être substitué par quelques appels de fonctions contribue à faciliter tests et maintenance. Par ailleurs, la partie de code correspondant au dialogue est "*mise en facteur*" : ainsi, la visualisation d'une fenêtre à l'écran s'écrit sous la forme d'une commande unique. Ceci améliore la lisibilité du programme, et, partant, sa

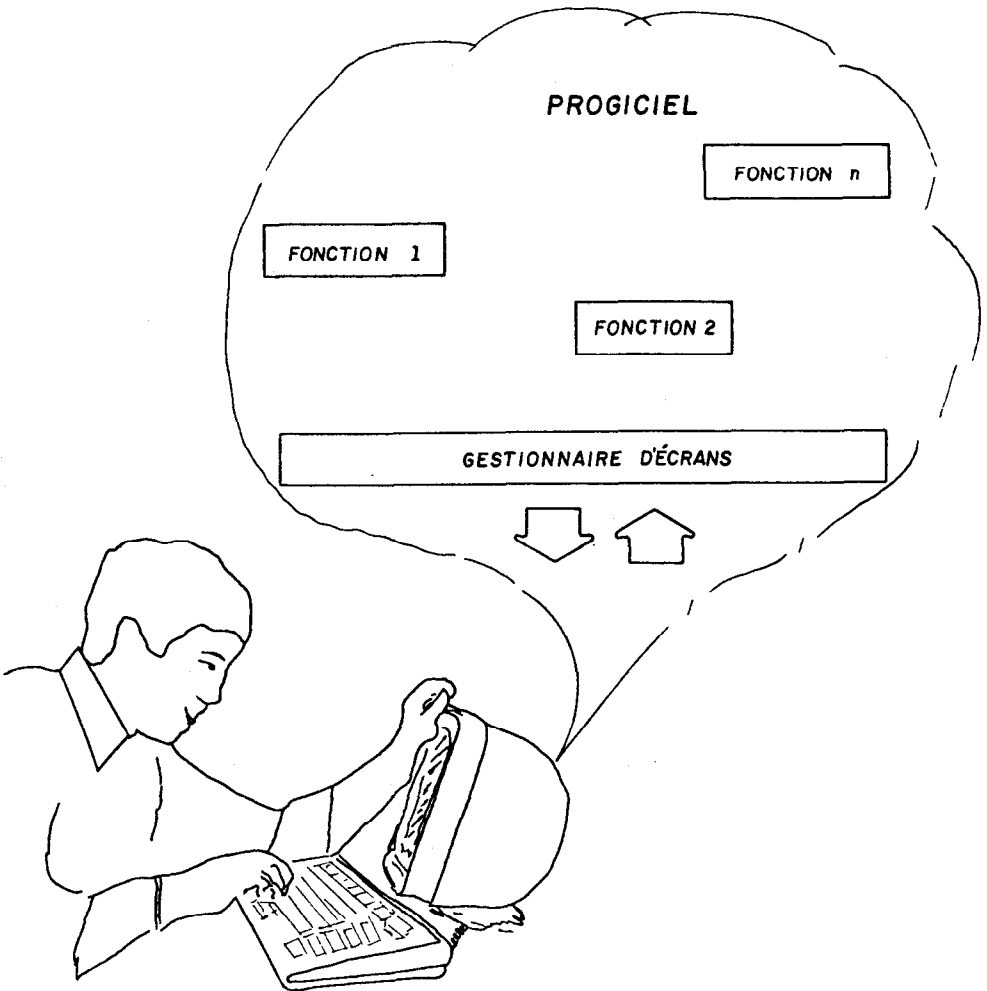


Figure 1 - Le gestionnaire d'écrans administre totalement la communication entre le progiciel et l'utilisateur.

maintenabilité. La portabilité de l'application augmente dans les mêmes proportions ;

- *d'automatiser une partie de la documentation.* Tout logiciel de création et utilisation d'écrans permet l'impression de ceux-ci, généralement sous deux formes distinctes. La première fournit l'image de l'écran tel qu'il sera visualisé au moment de l'exécution : au même titre que les lay-out de formulaires et les gabarits d'états, elle pourra faire partie de la documentation des entrées-sorties du progiciel, soumis à approbation du commanditaire avant de débiter la programmation. La seconde comporte en plus toutes les informations nécessaires sur le format et la position de chaque champ contenu dans l'écran en question : elle sera précieuse pour le programmeur chargé de la mise au point du programme qui l'utilise ;

- *d'améliorer le dessin des écrans.* Lorsque l'on ne dispose pas de cet outil, on doit dessiner les écrans sur un gabarit, de façon à déterminer la position (ligne et colonne) de chaque information et codifier les ordres de lecture et écriture correspondants. L'éditeur fournit avec le gestionnaire permet de dessiner les écrans interactivement, le plus souvent sous la forme selon laquelle ils apparaîtront en temps d'exécution, sans avoir à se soucier des positions des divers champs. Les produits les plus sophistiqués permettent d'améliorer la convivialité du progiciel, en tirant le meilleur parti des ressources offertes par les moniteurs : utilisation des attributs-vidéos monochromatiques (surbrillance, vidéo inversée, clignotement), des caractères spéciaux et semi-graphiques de l'IBM-PC (cadres, par exemple) ou des couleurs. Des efforts de programmation importants seraient nécessaires pour obtenir des couleurs. Des efforts de programmation importants seraient nécessaires pour atteindre un tel objectif sans l'utilisation de l'éditeur d'écrans ;

- *de viabiliser la séparation des tâches* d'analyse et de programmation. Lors de la mise au point d'un progiciel, le dessin des écrans peut être entièrement réalisé par l'analyste de spécification, indépendamment de la codification des programmes et de leur logique interne : ceci aura certainement pour effet d'améliorer le standard et le niveau de convivialité de l'application. En outre, la validation des données saisies peut également être prise en charge, au moins partiellement, par le gestionnaire d'écrans. Dans ces conditions, l'utilisation d'un tel produit s'accompagne d'un transfert substantiel d'activités de la phase de programmation vers la phase de spécification, ce qui est hautement bénéfique ;

- *d'optimiser les temps de réponse.* Dans la mesure où l'on utilise un produit spécifiquement conçu pour la manipulation des fenêtres et écrans, on peut espérer que les opérations correspondantes seront réalisées avec un niveau d'efficacité qu'il serait

difficile d'atteindre dans les conditions normales de développement d'application (langage de haut niveau, en particulier).

3. DESCRIPTION GENERALE

Pour la description générale des ressources d'un gestionnaire d'écrans, on est amené à définir trois contextes différents, liés au cycle de vie du système mis au point : la spécification, la programmation et l'exécution.

Dans le premier environnement, l'analyste utilise le produit pour dessiner les écrans et fenêtres qui interviendront dans le système applicatif projeté. Le gestionnaire est alors perçu comme un éditeur d'écrans, qui devra donc posséder des outils semblables à un éditeur de textes, tels que saisie de texte, mouvements du curseur, commandes de bloc, effacement, etc..

On pourrait à la limite se satisfaire d'un éditeur de textes standard (*Wordstar*, *Word*, *Wordperfect*, *Sprint*, etc.) pour réaliser les écrans. Cette solution apporterait toutefois quelques limitations dans la mesure où il faut créer une information qui ne peut pas être facilement véhiculée par le seul écran, comme les noms de champs, leurs types, leurs contenus par défaut, les valeurs initiales, minimales et maximales des champs numériques, etc..

Bien entendu, à l'instar des éditeurs de texte, l'éditeur d'écrans devra permettre de sauvegarder le résultat du travail sur disque (Figure 2), de le récupérer ultérieurement pour l'altérer si nécessaire, de l'exclure, de copier un écran pour en créer un nouveau, etc.. L'utilisation d'une bibliothèque de fenêtres et d'écrans pourra constituer une ressource intéressante, certains systèmes d'exploitation imposant une limite au nombre de fichiers contenus dans un volume ou dans une partition.

Enfin, l'éditeur devra permettre l'impression des écrans, sous les deux formes qui ont déjà été évoquées pour servir de documentation au programmeur et à l'utilisateur. La possibilité de créer un fichier texte contenant l'image de cette édition est appréciable pour permettre l'incorporation de cette documentation dans le cahier du programme et dans le manuel de l'utilisateur, par l'intermédiaire de l'utilisation d'un traitement de texte.

Dans le deuxième environnement, les programmes doivent être codifiés pour établir une interaction avec les écrans créés, qui se présentent désormais sous forme de fichiers sur disque, pour assurer le fonctionnement du progiciel.

Il faut alors disposer d'une bibliothèque de procédures ou de fonctions capables d'assurer, par simple appel en temps d'exécution, des tâches telles que la visualisation d'un écran ou d'une fenêtre, l'exhibition d'une information connue du système à

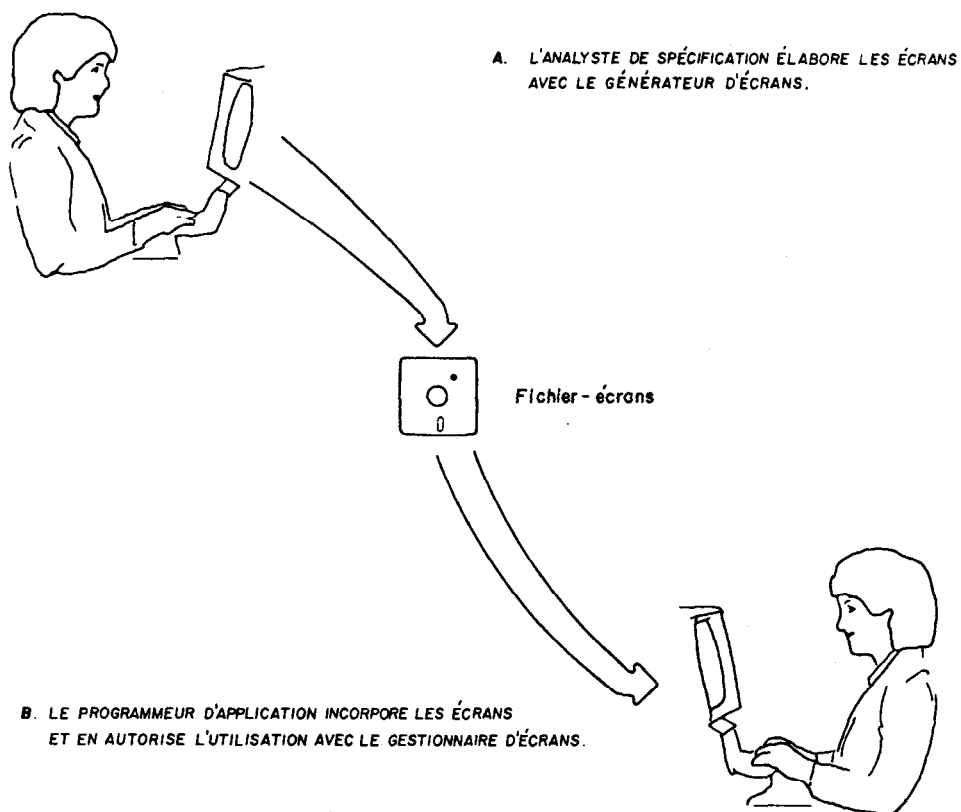


Figure 2 - L'utilisation d'un gestionnaire/générateur d'écrans met en évidence deux étapes distinctes.

l'intention de l'utilisateur (résultat d'un calcul, par exemple) ou encore la mise à disposition du progiciel d'une donnée fournie par l'opérateur.

L'interaction entre le programme d'application et les routines du gestionnaire d'écrans peut être réalisée de diverses manières, selon le langage et le produit utilisé. Dans le cas d'une version compilée d'un langage disposant de ressources d'appels de fonction et procédure d'une part, passant par une phase d'édition de liens d'autre part, le plus efficace sera de disposer des modules objets *relocables*, sous forme d'une bibliothèque complémentaire à concaténer avec la librairie standard du langage. Les routines du gestionnaire d'écrans seront alors automatiquement incluses dans le programme d'application.

Si l'on ne passe pas par une phase d'édition de liens (Turbo Pascal, par exemple), on préférera un produit composé d'un programme exécutable, chargé avant l'exécution dans un segment de mémoire distinct du segment de code et du segment de données (système d'exploitation MS-DOS). De cette façon, on aura les 64K du segment de code entièrement disponible pour le programme applicatif. Une autre solution, moins efficace, consiste à incorporer les sous-programmes du gestionnaire dans le programme source, puis compiler l'ensemble : l'inconvénient majeur de cette dernière solution est l'augmentation de la taille du programme objet, qui intègre alors toutes les routines (il y a néanmoins la possibilité d'utiliser le mécanisme de recouvrement).

Le dernier environnement est celui de l'utilisation effective de l'application développée, quand l'opérateur exécute les programmes qui composent le progiciel, et, par le biais de ceux-ci, manipule implicitement les fenêtres et écrans créés par le gestionnaire.

4. DESCRIPTION DE "EDITOR/TELADOR"

Le système d'administration d'écrans mis au point par l'équipe franco-brésilienne du programme SISGEO entre dans la catégorie des produits dont la description font l'objet de cette note. Il est utilisable en Turbo Pascal pour les systèmes d'exploitation CP/M 80 et MS-DOS. Rudimentaire, il convient bien pour servir d'exemple et illustrer dans le détail ces propos.

Comme son nom l'indique, il se compose de deux éléments distincts : un éditeur pour dessiner les écrans (telas, en portugais) et une bibliothèque d'accès appelée "*telador*".

4.1. Editor

L'éditeur est un programme qui permet la création et la maintenance de fichiers standardisés, contenant l'image et la définition des écrans qui seront utilisés dans l'application en cours de mise au point.

Pendant une session d'édition, on dispose de toutes les facilités pour dessiner la grille d'écran sur le moniteur : la définition des champs qui seront considérés comme variables (c'est à dire non connus lorsque l'on construit la grille, mais seulement déterminés en temps d'exécution du programme applicatif) est faite grâce aux caractères spéciaux placés sur la grille elle-même. Les conventions suivantes sont adoptées :

- un \$ représente le contenu d'un champ alphanumérique non protégé, c'est à dire accessible par l'utilisateur. En pratique, il s'agira donc d'un champ de saisie ;
- un # représente le contenu d'un champ numérique non protégé ;
- un ' représente le contenu d'un champ alphanumérique protégé (ou encore champ d'affichage alphanumérique), accessible seulement par le progiciel, et un ' représente le contenu d'un champ numérique protégé.

Tous les autres caractères rencontrés sur la grille sont considérés comme représentant des constantes : lorsqu'ils sont entourés de caractères spéciaux, ils sont entendus comme composant un masque d'édition.

Le dialogue offert par Editor est particulièrement simple et adopte le standard introduit par le traitement de textes Wordstar, pour tout ce qui concerne l'édition proprement dite. Dans le mode d'édition une aide en ligne est disponible pour les débutants.

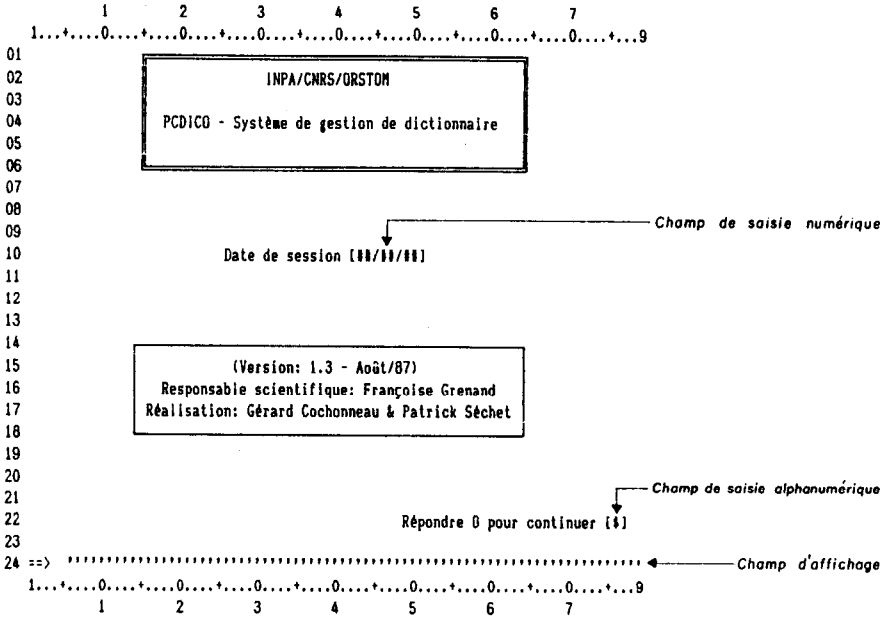
Les fichiers générés par l'éditeur contiennent, en plus de l'image de la grille d'écran, un dictionnaire recevant quelques informations sur les champs variables de l'écran : type, position et masque d'édition.

Un ensemble d'options permet d'administrer les "*fichier-écrans*" : création, mise à jour, sauvegarde, copie, changement de nom et impression. Cette dernière opération génère deux produits :

- la grille d'écran simple, image sur papier de ce qui sera présenté en temps d'exécution sur le moniteur ;
- la grille d'écran complète, qui joint à cette image le listing du dictionnaire des variables contenues sur celle-ci. Le programmeur dispose alors de la liste, du numéro et de la description de chacune de ces variables (Figure 3).

Micro-Editor, Copyright, 1986, by Nelio Domingues
 Map of: DICD0100.MAP

A. IMAGE DE LA GRILLE D'ÉCRAN



Directory of Variables

Number	Type	Line	Column	Length	!!!	Number	Type	Line	Column	Length
1	#	10	43	6	!!!	2	#	22	76	1
3	'	24	6	74	!!!					

B. DICTIONNAIRE DES VARIABLES

Figure 3 - L'impression sous la forme "écran + dictionnaire" est utile pour le programmeur.

4.2. Telador

Telador se charge de manipuler les fichier-écrans créés par Editor. C'est une bibliothèque (en code source Turbo Pascal : "book") de procédures à inclure dans les programmes applicatifs. Il autorise également la saisie des informations en fournissant certaines facilités élémentaires de manipulation de curseur et édition de champs (déplacement en avant et en arrière d'un caractère, d'un mot, ou d'un champ, insertion d'un caractère, exclusion d'un caractère ou d'un mot, mise à blanc d'un champ, etc.).

Telador permet de réaliser six fonctions : celles-ci peuvent être activées, par le programme-hôte, au moyen de commandes spécifiques qui se présentent sous la forme d'appels de procédures sans paramètre :

- la première, *OpenMap*, ouvre et lit le fichier-écran, rendant ainsi disponibles tous les paramètres des champs de saisie et affichage ;

- une autre routine, *ClearMap*, prépare le contenu des variables, protégées ou non, qui vont recevoir des informations, en changeant les caractères spéciaux (\$, #, ', ') par des espaces ;

- la procédure *WriteMap* réalise la visualisation effective de la grille sur l'écran : cette sous-routine est normalement appelée après les deux précédentes, de sorte que la grille apparait avec les champs variables en blanc. La réunion de ces trois sous-routines en une seule aurait amélioré la lisibilité du programme : on a préféré les dissocier pour augmenter les performances dans certaines situations (par exemple, quand une lecture préalable du fichier-écran est souhaitable, la routine *OpenMap* pourra être appelée seule) ;

- la saisie des informations est réalisée par le biais de la procédure sans paramètre *ReadVar*, quels que soient le numéro et le type des données à récupérer de la grille. Un index est utilisé, et mis à jour à chaque appel de cette routine, pour indiquer quelle variable est reçue par le programme-hôte. Lorsque c'est nécessaire, les commandes de mouvement du curseur ou d'édition de champ s'accompagnent automatiquement de la mise à jour de cet index, de sorte que la saisie est "full screen" ;

- la valeur courante de toutes les variables de la grille, qu'elles soient protégées ou accessibles à l'opérateur, peut être écrite sur l'écran par le biais d'une commande spécifique, *WriteVar*, sans qu'il soit nécessaire d'écrire de nouveau toute la grille sur le moniteur. Enfin, la procédure *ClearVar* remet toutes les variables internes du programme-hôte en relation avec le telador à blanc, ce qui permet de traiter facilement le cas fréquent des saisies répétées sur la même grille (plusieurs articles d'un même fichier : mode "input").

Quelques options sont disponibles, comme celle de donner ou non l'écho à l'écran d'un caractère taper au clavier, ce qui permet le traitement adéquat des informations réservées (mot de passe, par exemple), ou encore de transformer systématiquement en majuscules toutes les données saisies.

4.3. Exemple d'utilisation

La séquence de commandes présentée ci-dessous est typique de l'utilisation de Telador dans un programme-hôte, pour la saisie d'informations :

```

OpenMap ;          (* ouvrir et initialiser la grille *)
ClearMap ;        (* mettre à blanc les champs variables *)
WriteMap ;        (* visualiser la grille à l'écran *)
InformerChamps ; (* valeurs des champs protégés *)
WriteVar ;        (* écriture effective de ces valeurs *)
repeat            (* boucle sur les champs à saisir *)
  ReadVar ;       (* lecture de la variable courante *)
  ValiderChamps  (* critique du champ correspondant *)
until Confirme ;
WriteVar ;        (* écriture de tous les champs *)

```

OpenMap ouvre le fichier-écran qui lui est indiqué par l'intermédiaire d'une variable globale, contenant le nom de ce fichier. La procédure interne InformerChamps, écrite par le programmeur d'applications, est destinée à transmettre certaines données du progiciel à l'utilisateur, comme la date de session, par exemple. Celles-ci seront effectivement montrées à l'écran à la suite de l'appel de WriteVar.

La prise en considération des informations fournies par l'utilisateur final est réalisée à l'intérieur d'une instruction répétitive, ReadVar, qui assure à chaque appel l'incrémentement de l'index pointeur de champ. La procédure interne ValiderChamps, également écrite par l'utilisateur, doit fournir le traitement adéquat à chaque variable : ce pourra être une instruction "case" utilisant l'index pointeur de champ comme sélecteur.

Enfin, il est naturel d'incorporer, comme dernière donnée à saisir par l'utilisateur, un champ de confirmation dont le résultat est exprimé sous forme de booléen (Confirme), pour valider définitivement les informations fournies par l'opérateur. A l'issue de la saisie, un nouvel appel à WriteVar exhibe l'ensemble des valeurs des variables, telles qu'elles ont été enregistrées par le progiciel.

5. AUTRES GESTIONNAIRES D'ECRAN

L'utilisation de ce premier outil, par l'équipe franco-brésilienne du programme SISGEO lors de la mise au point de deux applications différentes, a confirmé tous les bénéfices que l'on pouvait en tirer, et qui ont été mis en évidence au début de cet article.

L'assimilation de cette technique par les programmeurs peut être considérée comme satisfaisante, bien que dans la plupart des cas ceux-ci tardent à percevoir tout le potentiel d'un tel outil. L'adoption d'un gestionnaire d'écrans plus complet et plus sophistiqué a donc été décidée. Celle-ci coïncide avec l'exclusion des micro-ordinateurs de 8 bits, sous CP/M, comme configuration minimale d'opération des modules locaux développés dans le cadre du projet.

Une étude rapide des progiciels disponibles susceptibles d'assurer la création, l'administration et l'utilisation des écrans et fenêtres dans l'environnement PC-DOS a donc été menée : ses résultats sont présentés ci-après.

5.1. Telador, version 2.0

Dans un premier temps on a cherché à mettre au point une nouvelle version de "Telador", essentiellement dans l'optique d'éliminer les inconvénients majeurs qui avaient été détectés lors de l'utilisation de la première version. Ces inconvénients sont les suivants :

- la syntaxe adoptée pour les fonctions utilisant les écrans avait été choisie pour apporter un maximum de souplesse : en contrepartie, les programmes s'en trouvent parfois un peu alourdis (une affectation et trois appels de procédure pour montrer un écran, par exemple) ;

- la taille des écrans est relativement importante, compte tenu de la présence du dictionnaire associé à la grille. La taille minimale est de 2K (25 lignes x 80 colonnes) : elle croît avec le nombre (et la taille) des variables contenues dans la grille, pouvant atteindre 4K. cela est surtout gênant dans l'environnement PC-DOS, quand l'espace disponible sur une unité de volume standard (une disquette) est de 360K ;

- le nombre des fichiers créés devient relativement important, dans la mesure où chaque grille d'écran constitue un fichier. Bien que le nombre des écrans créés varie surtout avec le "confort" que l'on prétend fournir à l'utilisateur, les deux applications qui ont été mises au point peuvent servir d'exemple, la première utilisant dix écrans (6.000 lignes de code source), la seconde près de soixante (45.000 lignes de code source). On remarquera que ce problème est cette fois plus sensible dans

l'environnement CP/M, quand le nombre maximal de fichiers par volume est nettement plus limité (64). De plus, un nombre de fichiers élevé allonge le temps d'installation de l'application et tend à confondre l'utilisateur final ;

- certaines opérations, comme la visualisation d'un écran ou l'affichage des champs sont relativement lentes. Le fait qu'elles soient écrites en langage de haut niveau est une des raisons de cette lenteur. De plus, comme on a cherché à éliminer au maximum les paramètres des procédures, l'appel à WriteVar écrit systématiquement tous les champs sur l'écran, ce qui est extrêmement peu performant.

Telador 2 simplifie le plus possible l'interface avec le programme hôte : une procédure unique est utilisée et est appelée avec trois paramètres : nom de l'écran, numéro de la variable à considérer et nom de zone de transfert. L'affichage d'une variable est ainsi réalisé en fournissant le champ à visualiser dans la zone de transfert : si la grille n'est pas présente à l'écran, elle est automatiquement visualisée. Le résultat d'une saisie est également obtenu dans cette même zone.

Pour réduire l'encombrement du fichier-écran, on a exploité le fait que la grille contient, en elle-même, toutes les caractéristiques des champs (format, position, longueur et masque sont fournis par les caractères spéciaux utilisés). Telador 2 utilise donc des fichiers de taille fixe (2K) et construit le dictionnaire de variables chaque fois qu'il est nécessaire.

Le problème de la multiplication des fichiers, causé par le nombre important d'écrans associés à une application, ne pouvait être contourné qu'en regroupant ceux-ci dans des bibliothèques. Telador 2 tire profit de ce que ses écrans sont de longueur fixe pour administrer efficacement des bibliothèques d'écrans constituées par des fichiers séquentiels indexés, offrant toutes les facilités de gestion (création de bibliothèque, inventaire, accès, modification, inclusion et exclusion de grilles d'écran).

Telador 2 offre naturellement une meilleure performance que la première version, ne serait-ce que parce qu'il utilise l'allocation dynamique. De plus l'affichage isolé d'un champ est autorisé, ce qui élimine le problème présenté par la procédure WriteVar. Enfin, on a pas voulu pour cette seconde version se préoccuper de la portabilité vers le micro de 8 bits, ce qui a permis d'utiliser la visualisation par adressage direct en mémoire, réalisée par appel au système d'exploitation DOS.

5.2. Masque

Il s'agit d'un générateur-gestionnaire de grilles d'écran conçu et mis au point par Gérard Cochonneau, de l'ORSTOM. Le module d'utilisation des écrans étant écrit en assembleur 8088, il est utilisable dans plusieurs environnements de programmation (Basic compilé, Fortran-77, Cobol et Turbo-Pascal), ce qui en constitue la caractéristique la plus intéressante.

L'éditeur de grilles d'écran fourni avec Masque s'appuie sur un procédé un peu différent de celle des autres produits présentés dans cet article. En effet, la grille est vue comme un ensemble de lignes, chacune d'elles pouvant contenir un ou plusieurs champs : de cette façon, l'éditeur n'offre et n'utilise aucune commande "*plein écran*". Dans ces conditions, plutôt que sur la grille d'écran proprement dite, c'est sur une fenêtre de définition de champ que l'analyste travaille pour dessiner et définir sa grille d'écran.

Un champ peut être constant ou variable. Pour définir une ligne d'écran ou n'apparaît ni champ de saisie ni d'affichage, on choisit le type fixe et on éditera alors la ligne entière : l'éditeur ignore la notion de zone ou de bloc. Pour définir un champ variable on utilise la même fenêtre en fournissant explicitement la position désirée (ligne et colonne du premier caractère à occuper par le champ). La ligne en question est alors automatiquement visualisée dans la fenêtre.

A chaque champ variable est associé un nom de champ, qui est utilisé par le programme d'application pour communiquer avec la grille : cette notion de nom de champ est très intéressante, dans la mesure où elle favorise grandement la lisibilité du programme (les fonctions de saisie et affichage devant invoquer explicitement le champ par son nom), en plus d'apporter une relative indépendance entre le programme et la grille. On regrettera toutefois que Masque ne réserve que trois positions à cet effet.

Le logiciel ne comporte pas la notion de type de champ (numérique, alphanumérique, etc.), ceci parce que les langages ne représentent pas tous les types de la même façon. Astucieusement, le type de champ est construit caractère par caractère, chaque position d'un champ ayant elle-même son propre type, choisi parmi A pour alphabétique, X pour alphanumérique, D pour numérique, S pour signe et E pour ASCII étendu (tous caractères imprimables). Cette ressource autorise la définition de champs mixtes (partie numérique et partie alphabétique, par exemple).

L'éditeur permet également d'associer un certain nombre d'attributs à chaque champ : masque d'édition, champ obligatoire ou non, champ à remplir totalement, valeurs initiale, maximale et minimale, ces dernières valables quels que soient les types des caractères qui composent le champ. Il est intéressant de noter ici que Masque ne fait pas de différence entre champ de saisie et

champ d'affichage (cette distinction ne présente en fait un intérêt qu'en mode de saisie pleine page, quand les champs d'affichage doivent être protégés d'une saisie intempestive).

L'éditeur permet bien entendu de visualiser la grille-écran telle qu'elle se présente à l'utilisateur final en temps d'exécution, et même de la tester, en simulant une utilisation réelle de la grille, pour vérifier l'adéquation correcte des attributs qui ont été définis pour les champs variables.

L'impression de l'écran est possible et fournit, avec l'image de l'écran, un tableau des caractéristiques de chaque champ variable contenu dans la grille. Enfin, le logiciel permet la conversion des écrans générés par un éditeur de texte quelconque, moyennant une opération dite de compilation et quelques restrictions concernant le format des fichiers d'entrée.

L'utilisation, dans un programme-hôte, des fichier-grilles générés par cet éditeur nécessite l'incorporation de Masque, comme module externe (appelé au moment de l'édition de liens, quand elle existe, ou déclaré "*external*"). La communication avec le module est réalisée par le biais des paramètres effectifs au moment de l'appel, en nombre variable selon la fonction désirée. La gestion de ce type d'interface variant d'un langage à l'autre, on se bornera à décrire l'utilisation de Masque dans un programme Turbo Pascal.

Sept fonctions au total sont autorisées : le code de la fonction à exécuter est passée au module à l'aide d'une variable globale (Op), sous forme d'une chaîne de deux caractères (par exemple RD pour Read, WR pour Write). L'appel de la procédure (Mask) doit présenter systématiquement deux paramètres, pour satisfaire les exigences de cohérence chères au langage Pascal. Toutefois, suivant la fonction appelée, l'un d'eux, et parfois les deux seront inutiles ("*dummy*").

La séquence d'instructions présentées ci-après est typique de l'utilisation de Masque dans un programme d'application :

```
Op := 'CL' ;                (* appel de la fonction Clear *)
Mask(Dummy,Dummy) ;       (* pour effacer l'écran *)
...
Grille := 'DEMO.' ;
Op := 'GT' ;              (* appel de la fonction Get *)
Mask(Grille,Dummy) ;     (* pour afficher l'écran *)
...
Nom := 'DAT.' ;
Valeur := '12/08/87' ;
Op := 'WR' ;              (* appel de la fonction Write *)
Mask(Nom,Valeur) ;       (* pour afficher un champ *)
...
```

```

Nom := 'TEL.' ;
Op := 'RD' ; (* appel de la fonction Read *)
Mask(Nom,Valeur) ; (* pour récupérer un champ *)
if Stat <> '00'
then Erreur ; (* test du code de retour *)

```

La fonction Get doit être utilisée pour afficher l'écran dans son état initial. Dans cet exemple, on utilise la fonction Write pour afficher le contenu de la variable "valeur" dans le champ indiqué par "nom". De la même façon, la fonction Read autorise l'opérateur à saisir le champ nom, et transmet le résultat au programme-hôte par le biais de la variable valeur. A l'issue de cette opération, la variable globale Stat, qui constitue le code de retour des fonctions de Masque, peut être testée pour vérifier le résultat de la fonction. En particulier, si une touche de fonction a été actionnée, Masque retourne le code correspondant au programme-hôte dans cette variable, ce qui permet de prévoir certaines actions (écran d'aide, abandon de la saisie, etc.).

Tant pour la fonction Read que pour la fonction Write, le logiciel permet de fournir plusieurs noms de champs concaténés dans le premier argument, ce qui autorise un affichage efficace des champs de l'écran, d'une part, la saisie en mode pleine page, d'autre part.

Trois autres fonctions sont fournies :

- Reset, pour remettre la grille d'écran dans son état initial (ou seulement un ou plusieurs champs variables), sans avoir à relire le fichier ;

- Display, pour visualiser un message d'information dans la vingt-cinquième ligne de l'écran ;

- Output Color, pour afficher une chaîne de caractères à partir d'une position quelconque de l'écran, avec le choix des attributs vidéo (et/ou de la couleur).

Les trois logiciels présentés ci-après sont des produits du commerce et, comme tels, fournissent des prestations plus complètes, essentiellement dans la partie d'édition. Ils permettront normalement de créer plus "confortablement" les écrans, ces derniers étant également plus soignés.

La description succincte qui en est donnée ci-après poursuit un objectif d'analyse comparative basée sur les fonctions d'un gestionnaire-générateur d'écrans : l'éditeur, en distinguant l'édition des zones constantes (le décor), la définition des champs (et les caractéristiques disponibles) et la gestion des grilles, d'une part, la bibliothèque d'utilisation, sa forme, sa syntaxe et ses ressources, d'autre part. Les fonctions d'édition de champ disponibles en temps d'exécution constituent un dernier critère, particulièrement important pour l'utilisateur final.

5.3. Display Manager

Ce gestionnaire d'écrans destiné à l'IBM-PC est un produit commercialisé depuis 1983 par Digital Research. Il a la particularité d'offrir, en plus des deux parties classiques d'édition et d'utilisation de grilles d'écran, un programme d'installation qui autorise le paramétrage des codes particuliers à chaque moniteur (taille de l'écran, positionnement du curseur, attributs vidéo, etc.).

L'éditeur est conçu pour générer des écrans numérotés, lesquels sont stockés dans un fichier appelé *"Display File"*. Plusieurs de ces bibliothèques peuvent être créées pour une application déterminée, chacune pouvant compter jusqu'à 250 écrans. L'éditeur distingue la partie constante des écrans (champs littéraux) de la partie variable, classée en champs d'entrée (de saisie) et champs de sortie (affichage).

Alors que la partie constante de l'écran ne peut recevoir aucun attribut vidéo sur un moniteur monochrome (surbrillance, vidéo inversée, soulignement, clignotement), ni couleur sur un moniteur polychromatique, les champs de saisie et d'affichage reçoivent un numéro de référence et l'éditeur permet de faire varier leurs attributs vidéo (resp. couleurs), pour en accroître le contraste et la visibilité. En fonction des caractéristiques du moniteur utilisé au moment de l'exécution, les attributs vidéo ou les couleurs définies pour le champ seront activés, ce qui évite le risque de mauvaise lisibilité lié à un choix unique.

Les ressources fournies par l'éditeur pour dessiner les écrans suivent sensiblement le standard Wordstar (mouvements du curseur, effacement, etc.). Il n'y a pas de commandes sur les blocs, mais des commandes spéciales d'édition de champ, préfixées <Ctrl>U, la position de ceux-ci devant être marquée sur la grille d'écran.

Pour définir toutes les caractéristiques des champs variables, Display Manager utilise le concept de *"Status Window"*, associant à chaque champ une fenêtre de paramètres qui doivent être fixés par l'analyste de spécification. En plus des attributs vidéo et couleurs, le choix d'un format spécifiant comment le contenu du champ doit être présenté dans l'espace qui lui est réservé (justification à gauche ou à droite, position du point décimal, code monétaire, etc.) est possible.

Dans le cas particulier des champs de saisie le choix d'un type entre alphabétique, alphanumérique (avec ou sans caractères spéciaux, avec ou sans écho, avec ou sans conversion en majuscules), entier ou réel est autorisé. L'information donnée par l'utilisateur en temps d'exécution est alors validée, caractère par caractère, en fonction de ce type. Le résultat provoqué par la saisie

d'un caractère interdit est également contrôlable par la "status window" de chaque champ (*beep*, fin de champ automatique), ainsi que le traitement (fin de champ ou caractère illégal) qui doit être réservé aux flèches de mouvement du curseur et aux touches de fonction.

Display Manager permet l'impression de chaque grille-d'écran avec le dictionnaire associé des champs d'entrée et sortie : cette impression peut être réorientée dans un fichier disque, le logiciel permettant la concaténation de celles-ci dans l'optique de faciliter la réalisation du manuel d'utilisation pour l'application en cours de mise au point.

Pour l'utilisation des grilles d'écran générées par l'éditeur, on dispose d'un certain nombre de fonctions. Le programmeur d'application doit alors en codifier les appels, le code propre aux fonctions étant inclus dans la phase d'édition de liens, à partir de la bibliothèque objet fournie par Display Manager. Ces fonctions doivent évidemment être déclarées comme externes, ce qui oblige à incorporer les déclarations correspondantes (fournies) dans le programme source.

Ce mode de fonctionnement interdit l'utilisation de ce produit par les versions interprétées (ou compilées mais sans étape d'édition de liens) des langages. De plus, la codification par fonctions (et non par procédures) en empêche l'utilisation en Cobol, langage qui n'accepte pas cette notion. Display Manager est néanmoins utilisable en Basic (CBasic), C (Digital Research), Pascal (Pascal MT+ 86) et PL1 (PLI-86).

Les fonctions de Display Manager retournent normalement une variable entière (code de retour), dont la valeur est zéro si l'opération s'est bien passée, une valeur négative dans le cas contraire. La séquence d'instructions ci-dessous est typique de l'utilisation de ces fonctions dans un programme d'application écrit en Pascal :

```
var
  Retour : integer ;
  Saisie : string[80] ;
...
Retour := OpnDis('EcranS.DIS') ;      (* ouvrir display file *)
Retour := DispD(1) ;                  (* montrer l'écran n 1 *)
Retour := PosF(3) ;                   (* choisir le champ 3 *)
Retour := PutF('Titre') ;            (* affichage du champ *)
Retour := NxtF(-2) ;                  (* positionner *)
Saisie := GetF ;                       (* récupérer l'entrée *)
```

L'appel à la fonction `InitDM`, dont le rôle est de fixer les caractéristiques du moniteur utilisé à l'exécution, doit être fait avant toute utilisation des autres fonctions. Une bibliothèque doit ensuite être ouverte et un écran particulier chargé : il n'est pas possible d'avoir plus d'un écran actif à la fois, ce qui est un facteur limitant pour l'utilisation de fenêtres dans une application.

L'affichage d'un texte dans un champ est assuré par la fonction `PutF`, tandis que le contenu d'un champ de saisie est récupéré (et validé) par l'intermédiaire de la fonction `GetF`, toutes deux agissant sur le champ courant. Le positionnement absolu est obtenu par `PosF`, le positionnement relatif par `NxtF`, dont l'argument permet plusieurs variations (champ suivant, champ précédent, d'entrée ou de sortie, dernier ou premier champ).

Une valeur défaut peut être donnée (par `PutF`) pour un champ de saisie, constituant de cette façon un guide pour l'utilisateur : la fonction `UpdF` devra alors être utilisée à la place de `GetF`, pour récupérer le contenu du champ même si l'utilisateur n'a entré aucun nouveau caractère.

Par ailleurs, `Display Manager` fournit quelques ressources sophistiquées pour modifier les attributs d'un champ ou tester les capacités du moniteur en temps d'exécution.

5.4. Screen Maker

Ce produit commercialisé depuis janvier 1987 dans sa version 1.3 par *"Logiciel et Médias"* est un éditeur d'écrans qui génère une procédure d'affichage en code source (Turbo Pascal), auquel est associé un ensemble de bibliothèques complémentaires autonomes, également en code source, facilitant l'utilisation des écrans et fenêtres.

`Screen Maker` s'accompagne d'un éditeur qui permet de créer ou modifier un écran, ou une fenêtre, à la manière d'un éditeur de texte standard, mais en proposant des fonctions supplémentaires de définition des champs d'entrée/sortie ou de génération et gestion de fenêtres. Les commandes de cet éditeur font un plein emploi des ressources du clavier (flèches, touches de fonction, etc.) pour produire les mouvements du curseur, l'effacement, le centrage, les commandes sur les blocs, le passage en mode insertion, etc..

Un mode spécial de modification de bloc permet de traiter globalement des parties d'écran, appelées blocs, permettant de copier, déplacer, d'effacer un bloc, en modifier et en contrôler la taille, etc.. Le bloc sert également de cible aux commandes appelées de *"boîtes et lignes"*, destinées à tracer des cadres et des lignes de séparation dans un bloc. Couleurs et attributs vidéo sont également gérés de façon très complète. Le mode dessin est une caractéristique intéressante de l'éditeur fourni par `Screen Maker` : il permet

de tracer une figure de forme quelconque, simplement en agissant sur les touches de direction du curseur.

La définition des champs d'entrée/sorties est réalisée durant la phase d'édition de l'écran : ils sont repérés par des caractères spéciaux, différents pour les champs de saisie et d'affichage, placés dans les positions désirées sur l'écran. Le type de champ, spécifié par une lettre située après le dernier caractère du champ à caractériser, est choisi entre *s* pour *string*, *i* pour *entier*, *r* pour *réel*, *t* pour *heure* et *d* pour *date*. Ces deux derniers champs, s'ils existent, provoqueront l'affichage de l'heure et de la date système aux endroits indiqués.

Les attributs vidéo et la couleur des champs peuvent être définis indépendamment, champ par champ : par contre, certaines autres caractéristiques (champ obligatoire ou non, écho ou non) devront être gérées par le programme-hôte.

Screen Maker associe l'impression de l'écran, et du dictionnaire de variables, à l'opération qui consiste à générer la procédure en code source destinée à l'administration de cet écran. Le choix du label de cette procédure est laissé à critère de l'utilisateur, ainsi que le nom du fichier qui la contiendra et qui devra être inclus dans le programme d'application.

Pour l'utilisation des écrans et fenêtres générés par l'éditeur, il est nécessaire d'inclure certaines bibliothèques fournies par le logiciel, en fonction des ressources utilisées :

- Video1.usr, doit toujours être incluse ;
- Windiws1.usr, correspond à l'affichage des fenêtres ;
- Keyboard.usr, Datetime.usr et Scrdrive.usr sont

nécessaires pour pouvoir gérer les champs d'entrée/sortie.

Screen Maker fournit une procédure spéciale, EditFields, pour traiter un écran globalement en éditant les champs d'entrée en mode pleine page. Pour une édition champ par champ on aura la séquence d'instructions typique suivante :

```
(*SI SCREEN.COD *)           (* fichier de code généré *)
...
InitVideo ;                   (* initialise l'adresse écran *)
TabScreen ;                   (* initialise les variables *)
Screen(0) ;                   (* visualisation de l'écran *)
EditFields ;
PutField(var.1.pl.p2) ;      (* affiche le champ 1 *)
GetField(var.2) ;           (* saisie du champ 2 *)
```

L'appel Screen(0) visualise l'écran SCREEN dans la page 0 : le logiciel permet l'utilisation des quatre pages de mémoire d'écran disponibles sur la carte graphique en mode 80 colonnes (procédure DisplayPage), d'où la possibilité d'un affichage instantané des

écrans contenus dans des pages différentes, puisque cette opération correspond à un simple changement de l'adresse de base du contrôleur d'écran.

La procédure `PutField` affiche le contenu de la variable (sans type) `var` dans le champ indiqué par l'entier qui constitue le deuxième paramètre de l'appel. Les deux paramètres suivants, sont des booléens qui indiquent certaines caractéristiques qui varient suivant le type de champ (centrage, passage en majuscules, signe d'un champ numérique, etc.).

La procédure `GetField` récupère le contenu du champ d'entrée indiqué comme second paramètre de l'appel, dans une variable sans type (`var`). L'appel des procédures standards `InitVideo` et `TabScreen` est nécessaire pour initialiser les opérations et le tableau des variables de l'écran.

Un certain nombre de fonctions complémentaires sont disponibles pour la gestion des champs : elles s'expriment sous forme de procédures contenues dans la bibliothèque `Scrdrive.usr`, appelables par le programme-hôte. De cette façon, on peut imposer des limites minimale et maximale aux champs de saisie numériques (`SetLowerLimit` et `SetUpperLimit`), imposer un masque de saisie pour l'entrée de chaînes de caractères (`SetMaskTo`), initialiser un champ de saisie avec une valeur défaut (`InitInputField`), manipuler les erreurs de saisie dans le programme d'application (`UserErrorRoutine`), etc..

On remarquera que cette manière de définir les attributs des champs transfère le travail de validation correspondant à la phase de programmation (il ne sera d'ailleurs pas nécessairement plus simple que s'il avait été codifié directement avec les instructions de base du langage), alors que quand ces mêmes caractéristiques peuvent être définies au moment de l'édition des écrans, le travail correspondant est réalisé en temps de spécification, ce qui paraît plus adéquat.

La bibliothèque `Datetime.usr` fournit les routines nécessaires à la mise à jour des champs correspondants à partir de la date et de l'heure du système.

5.5. Turbo Screen

Il s'agit d'un progiciel distribué par la société PC-Soft : la version 1.2 est datée de février 1987. Egalement destiné à la génération d'écrans et fenêtres, et à leur utilisation dans un programme d'application codé en Turbo Pascal, ce produit se distingue du précédent par le fait que la bibliothèque contenant les procédures de gestion des écrans constitue un module résident unique, `Affturbo.com`, qui doit être chargé en mémoire préalablement à l'exécution.

Cette solution offre un avantage déterminant sur la précédente, dans la mesure où le segment de code réservé au programme est entièrement disponible, alors que dans le cas de l'inclusion des routines en code source, cet espace (64K) se trouve sensiblement réduit (de 12K environ pour Telador, jusqu'à 26K pour Screen Maker en utilisant toutes les options disponibles dans ce dernier logiciel). La souplesse apportée en contrepartie, par l'inclusion de procédures en code source (modifiables par le programmeur), n'est normalement pas relevant, d'autant plus que toute modification sera extrêmement préjudiciable à la maintenance ultérieure des programmes.

L'éditeur d'écrans disponible avec Turbo Screen est particulièrement complet et offre l'essentiel des fonctionnalités que l'on peut attendre d'un éditeur moderne (aide en ligne, gestion des caractères semi-graphiques, tracé de cadres, coloriage de zones, paramétrage des touches de fonction, manipulation de fichiers et changement de "drive" ou de répertoire, etc.). Il est possible à tout moment de faire appel au système d'exploitation et de revenir à l'éditeur, sans incidence sur le travail en cours. Il offre la particularité de sauvegarder écrans ou fenêtres sous la forme de fichiers, ou sous la forme de bibliothèques (fichiers séquentiels, d'où une gestion plus rudimentaire, mais simplifiée).

La partie constante des écrans est traitée sous la forme de zones et il n'y a pas de distinction a priori entre les champs de saisie et ceux d'affichage. La notion de zone permet une manipulation commode avec les instructions disponibles de définition, déplacement, recopie, effacement, coloriage, tracé de cadre : ces zones peuvent de plus être sauvegardées isolément (fenêtres).

Pour la génération d'écrans "portables" couleur/monochrome, l'éditeur offre la possibilité de visualiser l'image-écran sur deux moniteurs connectés simultanément, ce qui permet de s'assurer de la lisibilité sur les deux types d'écran.

Les champs sont repérés par leur position sur la grille et se voient attribuer un nom de champ, qui sera utilisé par les commandes de saisie et affichage dans les programmes applicatifs. La définition des caractéristiques d'un champ se fait dans une fenêtre analogue à la "Status Window" de Display Manager, avec sensiblement les mêmes ressources : validation automatique ou non, champ obligatoire, saisie invisible, transformation en majuscules, divers types de validation et cadrages, couleurs, etc..

Dans le cas de champs numériques, entier ou réel, valeurs minimale et maximale peuvent être stipulées, la valeur entrée étant automatiquement contrôlée en fin de saisie. La cohérence d'un champ de type date (nombre de jours du mois, etc.) est également testée.

Enfin, ce progiciel fournit deux notions particulières : tableau de champs et message associé à un champ de saisie. Le tableau de champs est très utile dans certaines situations de capitalisation d'information, quand il simplifiera grandement la codification. Par ailleurs, un message peut être associé à chaque champ : il s'affiche automatiquement en 25ème ligne lors de la saisie, et s'efface dès validation (avec restauration du contenu antérieur de la ligne).

Turbo Screen permet l'impression de chaque grille-écran avec ou sans le dictionnaire : une impression particulière, de type "hard copy", reflète les effets spéciaux (attributs vidéo).

L'utilisation des grilles d'écran générées par ce progiciel est assurée par le module résident Affturbo.com, qui intercepte les ordres passés par le programme. Cette communication est réalisée à partir des fonctions classiques d'affichage, Write, et de lecture au clavier, ReadLn, identifiées par des caractères spéciaux (délimiteurs de début et fin de commande).

Le nom de la fonction à exécuter, d'une part, les paramètres de cette fonction, d'autre part, sont alors donnés sous forme d'une chaîne de caractères. Certaines de ces fonctions retournent une valeur (ou compte-rendu, CR), nulle quand tout s'est bien passé et qui peut être testée par le programme-hôte.

Turbo Screen autorise le traitement des fenêtres et écrans selon deux modes distincts : le mode champ à champ et le mode pleine page. Dans le premier mode le programme-hôte gère lui-même le parcours de champ à champ, en effectuant n'importe quel traitement après la validation de chaque champ (décodification et affichage d'un libellé à partir d'une table, ouverture d'une fenêtre, etc.). En mode pleine page, c'est le module résident qui gère seul les déplacements de champ à champ et le programme-hôte ne "reprend la main" qu'après validation de la totalité de la fenêtre. Cette dernière solution est moins souple, mais réduit considérablement le code source nécessaire : elle est bien adaptée au cas de saisie de fiches entières.

Le mode pleine page dispose d'un jeu de fonctions spéciales qui permettent la sélection des champs à saisir et l'ordre dans lequel ils doivent être saisis.

La séquence de commandes ci-après, qui réalise la visualisation d'un écran, l'affichage d'un champ (TVA), la saisie d'une variable (Prix) et son passage au programme principal, est typique de l'utilisation d'écrans dans ce progiciel, en mode champ par champ :

```

const
  fin = chr(01) ;
  ...
  Write(deb, 'UTILISE, ECRAN1.AID', fin) ;      (* montrer écran1 *)
  Write(deb, 'AFFICHE, TVA, =18.60', fin) ;    (* affichage *)
  Write(deb, 'SAISIE.Prix', fin) ;            (* saisie *)
  ReadLn(Prix) ;                               (* récupération *)

```

Alors que le délimiteur de fin est une constante, celui de début est une fonction qui permet d'adresser la recherche de la routine de Turbo Screen à exécuter dans le segment de mémoire contenant le module résident Affturbo.com.

Un grand nombre d'options sont disponibles, comme le "scrolling" de l'écran (fonctions MONTE et DESCEND), l'affichage de messages (fonctions MESSAGE et ERREUR), l'utilisation de la bibliothèque d'écrans (fonctions BIBLI et MEMO), la gestion des fenêtres en recouvrement (fonctions OUVRE, FERME et FUSIONNE), le stockage et la récupération de l'écran en cours (fonctions SAUVE et RETOUR), la définition de l'action des touches spéciales dans le contexte de saisie (SORTIE), le changement des couleurs en temps d'exécution, la gestion de menus, classiques ou déroulants, en tant qu'écrans spéciaux.

En temps d'exécution l'utilisateur final dispose pour la saisie des ressources d'édition standard, éventuellement modifiées par la fonction SORTIE : <Home>, <End>, -->, <-->, , <Ins>, <Esc> et <CR> ont une action conforme à l'expectative. En outre, si le mode pleine page a été défini, l'utilisateur pourra se déplacer à son gré à l'aide des quatre touches fléchées et de la touche <Tab>.

Enfin, Turbo Screen fournit trois ressources complémentaires intéressantes :

- un mode test (type "Trace On/Off") qui permet d'afficher sur la 25ème ligne chaque fonction demandée au module résident, dans le but de faciliter la débugation du programme ;

- la possibilité de fusionner deux écrans : celle-ci peut être utilisée pour introduire des caractères de grand format, correspondants à des polices fournies par le progiciel ;

- la possibilité d'effectuer un traitement parallèle (affichage de l'heure, par exemple) pendant la saisie des champs, moyennant l'utilisation d'une variable spéciale de compte-rendu, CR+, qui rend la main au programme-hôte tant que la fonction de saisie n'a pas été validée.

CONCLUSION

L'examen de plusieurs produits différents, pour la génération et l'utilisation de grilles d'écrans sur micro-ordinateurs PC-compatibles et dans des environnements de programmation généralement utilisés sur ce type de matériel, met en évidence un certain nombre de caractéristiques intéressantes. Celles-ci sont plus ou moins développées selon les produits et pourraient devenir des critères de choix entre ces logiciels.

Parmi les caractéristiques qui concernent l'éditeur proprement dit, on appréciera particulièrement la possibilité de création de fenêtres : les produits qui offrent la notion de zone (et cadres), comme Turbo Screen et Screen Maker, y sont bien préparés. L'édition de la grille telle qu'elle se présente à l'utilisateur final est une ressource minimale que tous les produits offrent.

La réalisation d'écrans très soignés et d'apparence professionnelle exige la maîtrise complète des attributs vidéo et de la couleur (Display Manager et surtout Turbo Screen et Screen Maker). Dans cet ordre d'idées, Display Manager est le seul produit permettant un paramétrage complet des ressources des moniteurs utilisés en temps d'exécution et, partant, autorise une visualisation optimale.

Les options disponibles à l'impression de la grille d'écran (avec ou sans le dictionnaire des champs variables) constituent un outil précieux pour la réalisation de la documentation du système applicatif mis au point.

La définition des champs à l'aide d'une fenêtre spécifique (Masque, Turbo Screen et Display Manager) est sans doute la solution la plus adéquate pour permettre de stipuler toutes les caractéristiques des champs de saisie et affichage. Celles-ci sont importantes dans la mesure où, lorsque les valeurs possibles des champs sont totalement spécifiées, la validation est prise en compte par le gestionnaire d'écrans, ce qui décharge d'autant la codification du programme principal.

Il est donc intéressant d'avoir la possibilité de préciser un maximum de paramètres, tels que format de champ, champ obligatoire ou non, valeurs défaut, minimale et maximale, etc.. Un traitement à effectuer sur un champ (cadrage, transformation automatique en majuscule, saisie invisible) peut également être défini, de façon à économiser du code dans le programme d'application : pour ce type de ressource, Turbo Screen est le plus complet, suivi de près par Display Manager et Masque.

En ce qui concerne l'utilisation des écrans, la notion de mode de saisie (pleine page ou champ par champ) est importante : elle est pourtant peu commentée par tous ces progiciels. Seul Turbo Screen met en oeuvre explicitement ces deux modes d'entrée, en

fournissant une procédure spécifique (Ecran) pour le mode pleine page.

A ce propos, la distinction entre champ de saisie et champ d'affichage, placée au premier plan par plusieurs produits (Editor/Telador, Display Manager et Screen Maker), n'apparaît guère importante qu'en mode de saisie pleine page. Turbo Screen et Masque dispensent facilement cette distinction, en autorisant la sélection des champs à saisir (et l'ordre de saisie) : transférer cette tâche au programme d'application ne paraît pas pénalisant outre mesure.

La plupart de ces logiciels ne permettent pas l'ouverture simultanée de plusieurs grilles (exception faite de Screen Maker) : en première analyse ceci devrait constituer un obstacle pour l'utilisation des fenêtres, ou du moins un facteur limitant (fenêtres relativement simples, sans champ de saisie).

La syntaxe des fonctions des gestionnaires d'écran est un sujet de préoccupation, du moins pour les utilisateurs (programmeurs d'application), dans la mesure où il ne semble pas en avoir été un pour les concepteurs. L'appel d'une procédure unique (Masque et Telador 2) pourrait être une solution : mais lorsque beaucoup de fonctions sont proposées, le problème des paramètres devient vite délicat et on confond rapidement ceux à fournir de ceux obtenus. Pour des raisons évidentes de lisibilité, les appels de procédure sont préférables à des appels de fonction (Display Manager). Dans ce domaine, un effort certain a été fait par Turbo Screen, avec des vocables tout à fait expressifs en français (certains le regretterons), comme UTILISE, AFFICHE, DESCEND, etc. : dommage que cet effort soit partiellement gâché par l'utilisation dans une chaîne de caractères, entre deux délimiteurs et à l'intérieur d'une procédure d'entrée/sortie standard de Turbo Pascal !

Le repérage des champs variables par des noms (Turbo Screen et Masque), plutôt que par des numéros, est de nature à améliorer substantiellement la lisibilité des programmes d'application.

L'espace occupé, en temps d'exécution, par le gestionnaire d'écrans constitue un facteur tout à fait relevant, principalement lorsque l'on ne dispose que d'un seul segment pour le programme. Cet espace croît bien entendu avec la panoplie de fonctionnalités offertes par cette partie du logiciel et, selon les versions, varie entre 12 et 26K. La solution du module résident, apportée par Turbo Screen, constitue un élément décisif pour les utilisateurs de Turbo Pascal, puisqu'il laisse intact le segment de code.

Le problème de l'espace occupé par les fichier-écrans n'est normalement pas aussi crucial : selon le produit, et le nombre de variables contenues sur la grille, la taille du fichier peut varier de 2

à près de 10K : seul Telador 2, qui ne stocke que l'image de l'écran, produit des écrans de taille fixe, et minimale, 2K. Pour cette raison, il offre une manipulation complète de bibliothèque d'écrans : Display Manager offre sensiblement la même capacité avec le "*Display File*", et Turbo Screen permet d'opter pour grilles isolées ou bibliothèque, ce qui constitue une particularité intéressante.

En ce qui concerne les ressources en temps d'exécution, les produits spécifiques à l'IBM-PC utilisent normalement les touches fléchées : seul Editor/Telador, opérationnel sur micro 8 bits, conserve le standard Wordstar. On notera de plus que Display Manager, Turbo Screen et Masque permettent le paramétrage des touches spéciales du clavier de l'IBM-PC, en définissant leur action en saisie (abandon, validation, champ suivant, champ précédent, etc.). Cette caractéristique permet d'augmenter la convivialité des applications mises au point.

L'utilisation d'un même générateur/gestionnaire de grilles d'écrans avec plusieurs langages de programmation différents n'est le privilège que de Masque et Display Manager, le premier couvrant pratiquement toute la gamme des langages les plus utilisés à l'heure actuelle sur les PC-compatibles. Cet aspect pourra constituer un facteur décisif de choix dans certaines situations, où plus d'un langage est utilisé dans une même application, parfois seulement pour des raisons historiques.

REFERENCES BIBLIOGRAPHIQUES

Digital Research : *Display Manager, productivity tool user's guide for the IBM Personal Computer Disk Operating System*, Pacific Grove, USA, July 1983.

PC/SOFT Informatique : *Manuel de référence Turbo Screen, pour les ordinateurs personnels IBM et compatibles*, Montpellier, février 1987.

P.Phillipot : *Manuel utilisateur Screen Maker*, Logiciel et Médias, Paris, janvier 1987.

G.Cochonneau : *Manuel d'utilisation de Masque*.

N.Domingues : *MicroEditor, manual do usuario*.

Domingues, N. & Séchet, P. : *Ferramentas para implementação de aplicações "tela-cheia"*.

LECTURE ET TRAITEMENT IN SITU SUR MICRO-ORDINATEUR DE CASSETTES DE DONNEES NUMERIQUES DE TERRAIN

Jacques Vassal
(Géophysique)

(UR 105, Milieu Physique et Environnement Climatique)

Centre ORSTOM
BP 1386
Dakar (Sénégal)

RESUME - Afin de pouvoir traiter immédiatement sur le terrain des données géophysiques enregistrées sur cassettes standards sous forme numérique, nous avons développé un logiciel de lecture et de traitement de ces cassettes sur micro-ordinateur. Après une brève description du format d'acquisition des données, on présente les différentes fonctions du logiciel : lecture et décodage de la cassette, traitement sur les fichiers et restitution du signal analogique sur imprimante.

INTRODUCTION

Lors de campagnes d'enregistrement d'un signal naturel en Géophysique (variations géomagnétiques, courants telluriques, ondes sismiques), il est important de pouvoir traiter sur place, au moindre coût, les données recueillies, et en particulier de visualiser le signal enregistré afin de détecter immédiatement toute anomalie de fonctionnement et de suivre l'évolution temporelle du signal.

Nous avons adapté et amélioré sur micro compatible IBM-PC un programme de lecture de cassette du laboratoire de Mosnier (CNRS) (sur HP-1000) en y ajoutant un module de restitution du signal analogique. Les logiciels graphiques que nous avons ne permettant pas le tracé en continu d'une série temporelle longue sur plusieurs mètres de papier, nous avons créé ce module de tracé en programmant directement une imprimante graphique matricielle.

La configuration minimale pour l'utilisation de ce logiciel est un micro-PC avec 256 ko et une imprimante matricielle.

1. ACQUISITIONS

Les chaînes d'acquisition sont constituées de deux capteurs magnétiques (H nord et D est) et de deux lignes telluriques de 500m chacune, orientées NS et EW. Le signal naturel ainsi recueilli est échantillonné et stocké sous forme numérique sur cassettes audio standards (60mn). La capacité d'une cassette complète est de 108.000 données (1800 blocs) dont le numéro de voie et la valeur numérique sont codés sur deux octets et groupés en blocs de 128 octets constitués de huit octets de label (horodateur, caractéristiques de la station) et de 2*60 octets de données. L'intervalle de temps (en seconde) entre deux blocs successifs dépend du nombre de voies utilisées V et de la cadence de scrutation C : $dT=60*C/V$. Suivant le nombre de voies scrutées et la cadence d'échantillonnage, l'autonomie d'une cassette varie de 6 heures à 75 jours.

Les données sont codées sur 12 bits, soit 4096 digits pleine échelle. Chaque bloc est formaté en binaire comme présenté sur la figure 1.

2. DEPOUILLEMENT DE LA CASSETTE

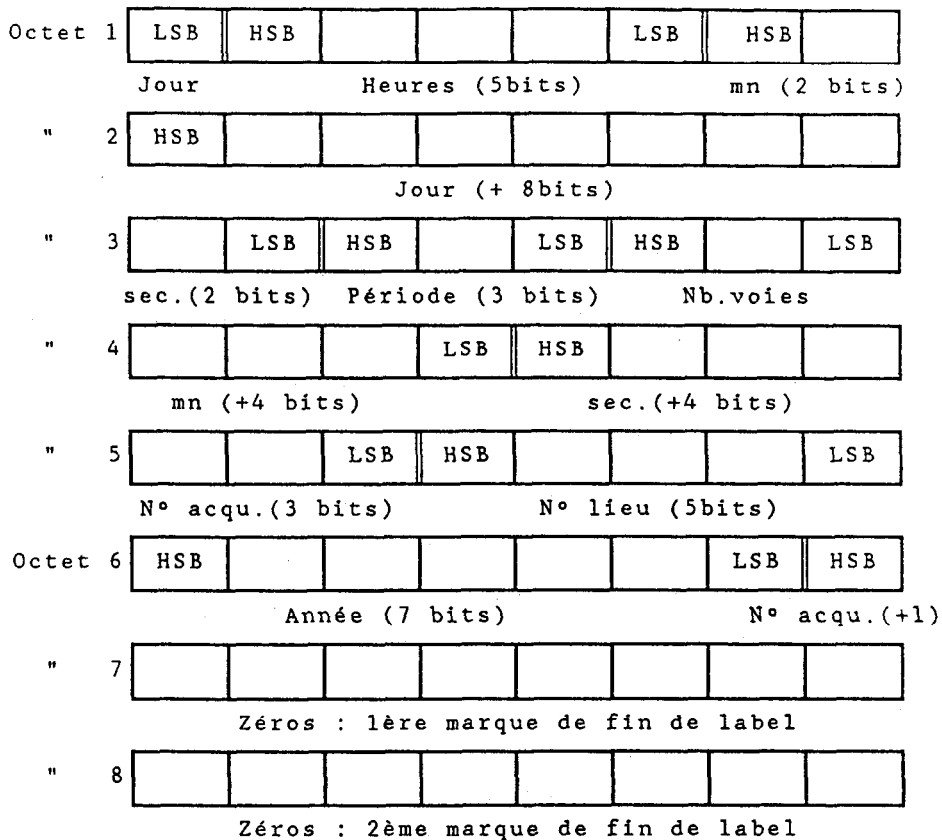
La chaîne de dépouillement se compose d'un lecteur de cassette Memodyne relié à un compatible IBM (PC, XT, AT) par une interface RS-232, et d'une imprimante matricielle.

Le logiciel est écrit en basic 2.01 sous forme modulaire, certains modules appelés par SHELL étant compilés (compilateur Basic 5.35) pour améliorer la vitesse d'exécution. Le lancement se fait à partir d'un fichier batch en tapant sous DOS : 'MTK7'. Il se présente sous forme de menus et de sous-menus, et est conversationnel. Un module de documentation donne toutes les indications pour la mise en oeuvre de la lecture de la cassette. Les principales options du programme sont les suivantes :

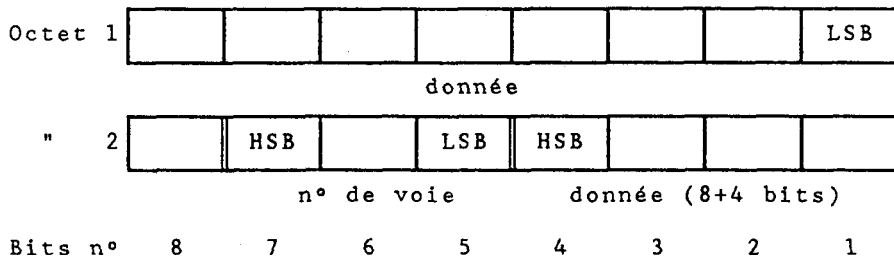
3. LECTURE DE LA CASSETTE

Lecture et décodage de la cassette : le programme lit la cassette à vitesse rapide avec transfert à 9600 bauds, recherche les marques de fin de label, décode les labels, vérifie la chronologie des blocs, décode les numéros de voies et les données associées, vérifie la séquence des voies, corrige par interpolation d'éventuels manques d'octets dus à des "trous" magnétiques, des poussières sur la cassette, ou des anomalies de transferts, et stocke les données décodées sur un fichier à accès direct constitué de blocs de 128 octets.

LABEL 8 octets :



DONNEES 2*60 octets :



LSB : Bit de plus bas niveau
 HSB : Bit de plus haut niveau

Une cassette complète de 1800 blocs est lue et décodée en 25 mn environ sur un XT.

Dans les deux premiers blocs (blocs en-tête) sont rassemblés en clair les caractéristiques de l'enregistrement : nom de la station, code d'acquisition, numéro de cassette, cadence d'échantillonnage, nombre de voies, amplifications des voies, date de début et de fin d'enregistrement, nombre de blocs données, ainsi que 128 caractères de commentaires. Tout ces renseignements sont entrés au clavier au démarrage de la lecture dans une zone d'acquisition protégée.

Les blocs données sont constitués d'une chaîne alphanumérique B\$ de 8 octets de label (horodateur) et de 2x60 octets de données (Valeur entière sur deux octets), les voies étant en séquence (d1V1, d2V1,..., dnV1, d1V2,..., dnVv).

Les valeurs numériques i sont obtenues :

- en Basic par la fonction $CVI(MID$(B$,Ki,2))$;
- en Fortran par $ICHAR(B(Ki))*256+ICHAR(B(Ki+1))$;

avec $Ki = 7+2i+2*(V-1)*60/NV$, où V est le numéro de la voie et NV le nombre de voies.

4. TRAITEMENT DES FICHIERS DECODES

Un menu d'utilitaires sur les fichiers décodés permet les premiers traitements suivants :

- correction des deux blocs d'en-tête ;
- lecture et listage des labels des blocs ;
- lecture et listage des données ;
- correction de quelques données ;
- retrait d'une tendance de la forme $aT+b$ (en cas de dérive d'une voie) sur une voie et entre deux valeurs choisies ;
- filtrage mathématique passe-bande d'un fichier avec création d'un fichier de données filtrées, de même structure que le fichier source. On utilise pour ce filtrage l'algorithme de récursion de Shanks ;
- hodogramme de deux voies.

5. DIFFERENCE DE DEUX FICHIERS

Ce module lit les deux fichiers dont on veut faire la différence, vérifie leur compatibilité, leur synchronisme et crée un nouveau fichier de même structure que les fichiers sources, constitué des différences valeur par valeur.

Afin de faciliter le tri et la recherche des différents fichiers, leurs noms sont constitués du nom de la station et d'une extension donnant le type de fichier : .dec, .dif et .fil pour les fichiers bruts, différences et filtrés.

6. GRAPHE DES VOIES

Ce module permet de tracer simultanément en continu les différentes voies sur imprimante. Les traces sont automatiquement centrées et les paramètres d'impression (valeurs d'échelles en amplitude et en temps) modifiables par l'utilisateur. Une option permet, en cas de dynamique importante, le rattrapage automatique des voies. Le graphe se fait entre deux dates (jour, heure, minute) données. Les valeurs d'échelle en amplitude sont indiquées, et les heures rondes marquées sur l'axe des temps. La définition en amplitude (axe des Y) est de 900 points pour 180 mm (sur papier en 240). Suivant le pas choisi pour le tracé, la restitution du signal peut se faire sur plusieurs mètres de papier en continu.

Ce module permet évidemment le tracé aussi bien du signal brut que des différences ou du signal filtré. Ce module compilé permet une sortie très rapide du graphe.

Les fonctions utilisées pour la programmation de la tête d'imprimante (espacement de ligne, programmation des aiguilles de la tête) sont communes à de nombreuses imprimantes matricielles à 8x8 aiguilles. Nous avons testé ce module avec plusieurs types d'imprimante (Oki 182 et 293, Citizen 120D, Epson), sans problèmes.

CONCLUSION

Notre logiciel permet de lire immédiatement au camp de base les cassettes recueillies aux stations de mesure et de faire les premiers traitements de signal (graphe, différence, filtrage). On peut ainsi modifier, en fonction des résultats obtenus, l'implantation des stations de mesure suivantes, ou revenir sur une station en cas d'anomalie. On évite ainsi toute perte de temps en optimisant au mieux la gestion des points de mesure.

Il permet de remplacer sur le terrain du matériel spécial, cher et encombrant, de lecture de cassettes (baie de lecture IFELEC) par un matériel informatique standard bon marché.

D'autre part, les disquettes obtenues peuvent être traitées complètement dès le retour au laboratoire, avant même la fin de la campagne de mesure. Les fichiers sont en effet directement utilisables par nos programmes Fortran d'analyse et de modélisation, eux aussi adaptés sur micro.

On mesure les progrès obtenus par cette technique quand on sait qu'il nous fallait deux ans pour obtenir les mêmes résultats du temps de l'enregistrement analogique nécessitant une numérisation en France, et encore quatre à six mois avec le dépouillement des cassettes sur HP-85 et les problèmes de compatibilité des fichiers au format IBM que ce système amenait.

PRESENTATION D'UN GESTIONNAIRE DE DONNEES NUMERIQUES HIERARCHISEES DESTINE AU DE- POUILLEMENT D'ENQUETES

Jacques Vaugelade & Marie Piron
(Démographie et Statistique)
(UR 702)

Centre ORSTOM de Ouagadougou
B.P. 182
Ouagadougou (Burkina Faso)

RESUME - Le dépouillement d'enquêtes est la suite d'opérations qui conduit du questionnaire au tableau de fréquence (appelé aussi tableau statistique, croisé ou de contingence). Cette suite d'opérations comprend des modules obligatoires : description des variables, description des contrôles, saisie et correction, calcul des nouvelles variables et tabulation.

Le questionnaire présenté ici offre la particularité de traiter des questionnaires hiérarchisés et de permettre une saisie interactive. Il peut être complété et interface avec des logiciels statistiques qui réalisent des opérations du type régression, analyse de données multidimensionnelles, etc..

Dans ce cas la structure hiérarchique doit être abandonnée au profit d'une structure rectangulaire qui entraîne la répartition des informations des niveaux supérieurs pour chaque unité des niveaux inférieurs.

1. OBJECTIFS RECHERCHES

. Concevoir un gestionnaire le moins limité possible dans le nombre d'enregistrements, des variables, des individus et dans les possibilités de composer la tabulation.

. Saisir les informations, aussi complexes soient-elles, dans l'organisation du questionnaire qui se présente parfois sous la forme de différents niveaux d'enquête. Il s'agit d'enregistrer les données sans redondance des unités supérieures et pour un nombre quelconque de niveaux (principe d'une structure hiérarchisée).

. Etre attentif aux détails qui facilitent ou permettent d'exécuter des opérations qui font parfois défaut sur les autres logiciels de même profil.

. Enfin, concevoir un logiciel pour micro-ordinateur afin de pouvoir saisir et traiter immédiatement l'information localement, suivant les exigences précitées, aspect non négligeable pour les études portant sur les pays en développement.

2. PRESENTATION DU GESTIONNAIRE

Ce gestionnaire, conçu par J.Vaugelade, est écrit en Basic pour micro-ordinateur compatible IBM. Il permet actuellement de traiter uniquement des données numériques entières (valeurs comprises entre -32 768 et +32 767).

Il se présente sous la forme de menus déroulants et comporte les modules essentiels à son fonctionnement, à savoir :

2.1. Description des unités, des variables et des contrôles

Cette phase préliminaire consiste à décrire la structuration de l'enquête (un questionnaire peut être composé d'un ensemble de sous-questionnaires définissant chacun des unités secondaires, tertiaires, etc.. L'unité principale est l'unité statistique d'enquête dont peuvent dépendre plusieurs unités de niveau 2, 3, etc.). Il s'agit de définir les types d'unité et leur niveau dans la hiérarchie, de constituer les informations nécessaires pour décrire les variables qui leur sont affectées (libellé, nom abrégé, valeurs possibles) et d'établir des contrôles croisés entre les variables, basés sur des tests logiques.

2.2. Saisie, mise à jour, contrôle des données

La saisie est effectuée de manière interactive avec contrôle et correction immédiate.

Elle se particularise par une structure hiérarchique qui offre par là même deux avantages : une saisie simple compte tenu de la complexité du questionnaire pouvant faire intervenir différents niveaux d'enquête et l'occupation du minimum de place dans les fichiers d'enregistrement.

Lors de la première saisie, il suffit d'appeler l'unité désirée en respectant l'ordre de la hiérarchie, les variables concernées s'affichent alors.

Avec le programme de mise à jour, il est possible de se déplacer dans le fichier afin d'effectuer des corrections sur les variables, de supprimer ou d'ajouter des variables ou des unités. Ceci permet, par conséquent, d'enregistrer plusieurs passages d'enquête en ajoutant soit des variables, soit des unités statistiques d'enquêtes.

Cette saisie est simultanément accompagnée d'un prétraitement permettant de contrôler la qualité de l'information en décelant les erreurs de codification ou de saisie. Un premier

contrôle nous assure que chaque réponse est possible car elle appartient à une catégorie de codes précisés (Cf. description des variables), le deuxième contrôle, dit croisé, permet de vérifier la pertinence de la valeur de la variable (Cf. description des contrôles).

2.3. Création des variables

A partir des variables existantes, il est souvent nécessaire de créer par synthèse d'autres variables plus adaptées aux objectifs de l'étude. Par cette fonction, il est possible de synthétiser les informations d'un niveau inférieur ou d'utiliser les informations des niveaux supérieurs (on peut de plus faire intervenir dans le calcul les unités précédentes appartenant au même type d'unité de la variable concernée). Il est également prévu de créer une variable pour une sous-population constituée à l'aide de filtres.

2.4. Tabulation

Ce module permet de composer des tableaux de contingence (ou de fréquence) à partir d'une ou de plusieurs variables. Il permet de recoder ou de reclasser une variable, de lui affecter un intitulé, d'en sélectionner ses modalités, de ventiler un nombre quelconque de variables avec le choix de faire intervenir le total ou le rebut, de pondérer la population et enfin de travailler sur un type de population précis, sélectionné par des filtres.

Cependant, les types d'unités de même niveau sont indépendants entre eux. Par conséquent, les informations d'une unité peuvent être corrélées soit entre elles, soit avec les informations des unités supérieures dont elles sont dépendantes. Mais rappelons que nous sommes toujours à même de synthétiser l'information de deux types d'unités de même niveau au type d'unité supérieure commune.

D'autres modules existent comme lister ou trier des données suivant des variables prédéfinies, récupérer des fichiers endommagés et accéder à des fichiers séquentiels. En effet, concernant ce dernier point, il est à préciser que la structure hiérarchique des fichiers impose un enregistrement à accès direct.

La sortie séquentielle en caractères peut s'effectuer sous la forme de tableau rectangulaire ou hiérarchique. Ce module a toute son importance puisqu'il permet de passer sur des logiciels statistiques spécifiques. A cet effet, un passage vers STATITCF est actuellement en cours de réalisation.

3. EVALUATION

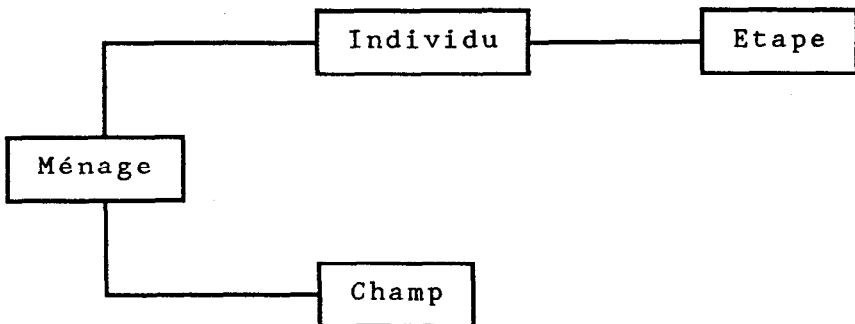
3.1. Propositions d'amélioration

- . Etendre au traitement de données non entières et non numériques.
- . Elargir les contrôles croisés, lors de la saisie, à des contrôles portant sur plusieurs variables, calqués en plus des tests logiques, sur le principe de création des variables.
- . Perfectionner l'ergonomie et notamment au niveau de l'homogénéité dans les différents programmes.
- . Optimiser la vitesse d'exécution des différents modules et du stockage des données.
- . Réaliser une version compilée.
- . Améliorer la présentation des tableaux croisés.
- . Et pourquoi pas, étudier les possibilités d'une manipulation directe sur le tableau des données par le déplacement du curseur.

3.2. Points forts

L'originalité de ce gestionnaire est de pouvoir traiter des fichiers hiérarchisés à un nombre quelconque de niveaux. Lorsque les fichiers sont limités à deux niveaux, il est toujours possible de les ramener à un niveau unique en prévoyant le nombre maximal d'occurrences du deuxième niveau. Cela n'est plus possible au delà de deux niveaux. Nous attirons donc l'attention sur le principe d'une saisie hiérarchique qui consiste à dégager du questionnaire de base plusieurs sous-questionnaires correspondant chacun à un type d'information recueillie et pouvant, par conséquent, être considérés comme des sous-unités statistiques.

Si nous envisageons un fichier avec quatre types d'unités (Ménage, Individu, Etape, Champ) avec la structure suivante :



Chaque unité est caractérisée par un certain nombre de variables et dépend impérativement d'une unité supérieure qui l'identifie. Les unités secondaires sont enregistrées autant de fois

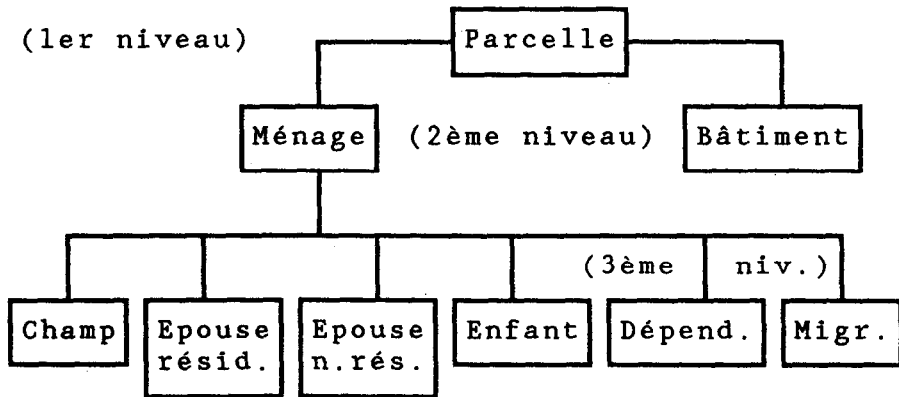
qu'il est nécessaire pour le questionnaire, en respectant l'ordre de la hiérarchie. Cette démarche permet, par conséquent, de focaliser progressivement les informations caractéristiques du ménage vers l'unité étape. Cette méthode de saisie présente l'avantage de traiter une catégorie d'enquête qui sera plus riche que les enquêtes restreintes à un seul type d'unité qui synthétisent et donc perdent l'information des sous-unités.

Ce gestionnaire offre un dimensionnement des fichiers en fonction des besoins de l'utilisateur, la seule limite relative reste la mémoire disponible de l'ordinateur, à savoir 64 Koctets pour le programme et les données.

Une grande souplesse d'utilisation est apportée par de nombreux outils de manipulation qui facilitent le traitement des données dont on augmente les possibilités (étude sur une sous-population sans revenir au tableau des données, existence de rebut, maniement des variables et de leurs modalités, etc.).

4. APPLICATION

M.Piron a utilisé et testé ce gestionnaire de données numériques pour une enquête socio-urbaine à passages répétés, comprenant neuf types d'unités réparties sur trois niveaux :



Elle n'a été contrainte à aucun dimensionnement, bien que le nombre de variables soit relativement important, plus de 300 variables et environ 13 000 unités confondues, après le deuxième passage de l'enquête. Elle a pu supprimer, modifier les variables, réaliser un premier traitement de données avant de passer sur des logiciels spécifiques tels que l'analyse multidimensionnelle, pour une étude plus approfondie des phénomènes urbains à Ouagadougou.

ANNEXE

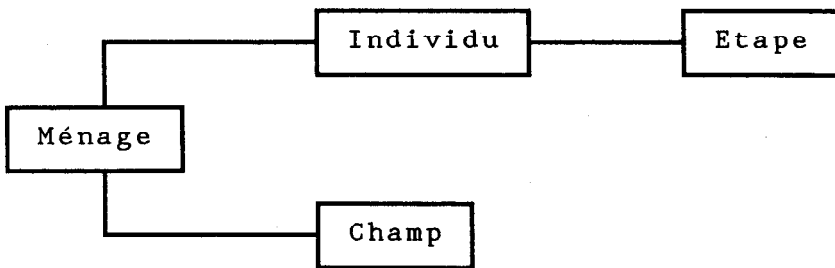
Structure des fichiers hiérarchisés : principe d'enregistrement

. Pour permettre la correction des unités et la mise à jour, le fichier doit être à accès direct, il faut alors choisir une longueur d'enregistrement. Plusieurs possibilités s'offrent, choisir l'unité ayant le plus de variables pour fixer la longueur de l'enregistrement conduit à une perte de place importante pour un micro-ordinateur ; créer un fichier par type d'unité présente deux inconvénients, le premier est une limitation du nombre d'unités par le nombre de fichiers pouvant être ouverts simultanément, le deuxième est une perte de temps en écriture et en lecture, car le temps de positionnement pour accéder à un enregistrement (secteur) est de l'ordre de cent fois le temps de lecture d'un secteur.

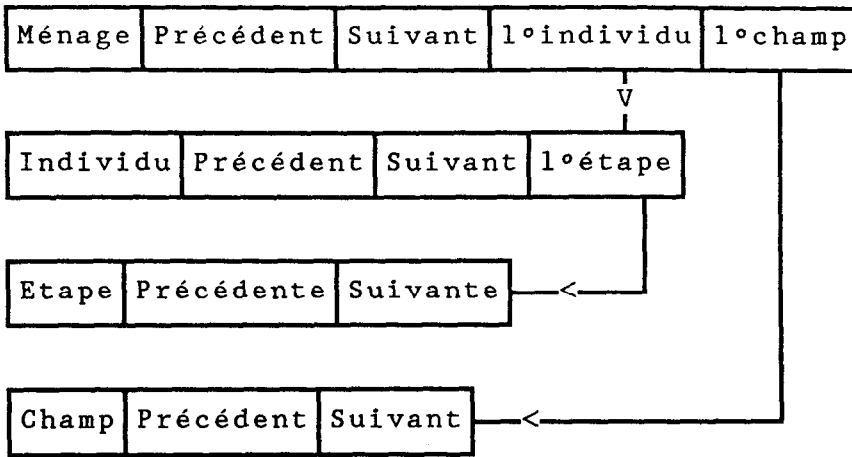
La structure retenue est donc une structure séquentielle, chaque enregistrement à une longueur fixe (par exemple 126 variables entières sur deux octets) et les unités sont mises les unes à la suite des autres.

. Pour permettre la mise à jour, il faut pouvoir se déplacer dans le fichier dans n'importe quel sens, vers l'avant ou vers l'arrière. On est donc conduit à introduire des pointeurs que l'on va illustrer par un exemple.

Soit un fichier avec quatre type d'unités, ménage, individu, étape, champ, avec la structure suivante :



Pour chaque unité on a un pointeur sur l'unité précédente et sur l'unité suivante, et pour chaque unité non terminale autant de pointeurs qu'il y a d'unités directement dépendantes.



Au cas où un pointeur est sans objet, sa valeur est zéro. Chaque pointeur donne le déplacement relatif, sauf si l'unité correspondante est dans le fichier des ajouts, auquel cas c'est le numéro de l'enregistrement (précédé d'un signe négatif) qui est le pointeur.

**Session 2 :
CONSTITUTION DES BASES
DE DONNEES**

**LA BASE DE DONNEES DES RESSOURCES GENE-
TIQUES DES CAFEIERS AFRICAINS EN COLLECTION
EN COTE D'IVOIRE**

François Anthony
(Génétique)
(UR 507)

Centre ORSTOM de Montpellier
B.P.5 045
34032 Montpellier Cédex (France)

RESUME - Le Centre des ressources génétiques des caféiers africains, constitué en Côte d'Ivoire par l'ORSTOM et l'I.R.C.C. (Institut de Recherche du Café et du Cacao), est unique au monde par le nombre d'espèces représentées, l'importance des effectifs et la diversité des origines. Plus de 7500 géotypes sauvages appartenant à une quinzaine d'espèces de *Coffea* ont été collectés dans environ 250 populations naturelles réparties dans huit pays d'Afrique. Les collections sont installées dans deux stations de recherche présentant des conditions écologiques différentes. Les programmes de recherche basés sur ce matériel végétal sont effectués à l'aide de descripteurs nombreux et variés.

L'informatisation des données a commencé à la fin des années 70 par la définition d'un système de numérotation des géotypes applicables aux caféiers sauvages et à leurs hybrides des deux premières générations. En 1985, les deux stations de conservation ont été équipées de micro-ordinateurs Goupil 3PC fonctionnant sous MS-DOS. Le support physique des données est formé par des disquettes 5" (360 Koctets). Les logiciels utilisés pour la gestion et les analyses statistiques ont été développés par Noirot, Déjardin, Mullon et Savy (N.D.M.S.) en Sbasic (Publication ORSTOM, à paraître).

La base de données conçue a pour objectifs de :

- permettre la gestion des caféiers plantés en relation avec leur origine géographique,
- constituer un répertoire du matériel végétal disponible,
- standardiser les descripteurs utilisés dans les études génétiques et agronomiques,
- faciliter l'accès et l'exploitation des données de l'évaluation.

Les données codées concernent la numérotation des individus, leur provenance, leur plantation en collection et leur évaluation. Les données de l'évaluation, actuellement saisies, sont relatives à l'intensité des floraisons, aux productions par date de récolte et aux caractéristiques des fruits et des grains (fertilité, technologie...).

Il reste à coder les descripteurs de la morphologie des caféiers et des caractères enzymatiques révélés par électrophorèse.

Les données sont stockées dans deux types de fichiers :

- les *fichiers_des_individus* sont constitués uniquement de variables alpha-numériques représentant les codes des génotypes, de leur origine et de leur mise en collection (une ou deux positions). Ils sont découpés par espèce. L'ensemble de ces fichiers forme un répertoire permettant la gestion des collections.

- les *fichiers_de_l'évaluation* contiennent d'une part des variables alpha-numériques pour le code des caféiers et leur position en parcelle et d'autre part les variables numériques de leur évaluation. Pour chaque type de données (floraisons, récoltes...), des fichiers différents sont créés par parcelle, espèce et année.

La place mémoire totale occupée est d'environ 3,5 millions d'octets dont la moitié pour les fichiers des individus. Le logiciel N.D.M.S. réservant systématiquement huit octets par variable, il serait possible de diminuer des deux tiers la taille du répertoire si le programme de création de fichier tenait compte du format des champs.

Tous les fichiers sont triés sur le code des caféiers et exploités séquentiellement.

Les problèmes rencontrés proviennent du matériel et du logiciel. Il est évident que la capacité mémoire des disquettes est insuffisante. Pour le moment, les fichiers sont répartis sur 28 disquettes. Seul l'emploi d'un disque dur évitera les nombreuses manipulations de disquettes. Le logiciel N.D.M.S. n'a pas été conçu dans le cadre d'une base de données. Il ne permet d'exploiter qu'un seul fichier à la fois et les fonctions sont donc limitées. C'est pourquoi il est nécessaire d'utiliser un autre logiciel de gestion, plus performant, et de transformer ensuite les fichiers pour accéder à la bibliothèque de programmes statistiques N.D.M.S. qui est bien adaptée aux besoins des biologistes.

Les principales perspectives résident dans l'emploi d'un autre micro-ordinateur (Compaq 286) avec un disque dur (20 M. octets) à Montpellier. Les données seront prochainement transférées pour utiliser le logiciel dBASE III. Tous les fichiers pourront alors être indexés sur le code des caféiers. Cela permettra de constituer un système offrant de meilleures conditions d'exploitation.

INTRODUCTION

Le Centre des Ressources Génétiques des caféiers africains, progressivement mis en place en Côte d'Ivoire par l'ORSTOM et l'I.R.C.C. (Institut de Recherche du Café et du Cacao), est unique au monde par le nombre d'espèces représentées, l'importance des effectifs et la diversité des origines. Plus de 7500 géotypes sauvages, appartenant à une quinzaine d'espèces de *Coffea*, ont été collectés dans environ 250 populations naturelles réparties dans huit pays d'Afrique. Les collections vivantes sont installées dans deux stations de recherche présentant des conditions écologiques différentes : au Mont Tonkoui (1100 m.), près de Man, pour les caféiers d'altitude et à Divo pour les caféiers de basse altitude. La diversité morpho-physiologique de ces caféiers spontanés et leur valeur agronomique sont étudiées à l'aide de descripteurs nombreux et variés dont certains, utilisés pendant plusieurs années, sont communs à la plupart des espèces.

L'informatisation des données a commencé à la fin des années 70 par la définition d'un système de numérotation des caféiers. Depuis 1985, les deux stations de conservation des ressources génétiques sont équipées de micro-ordinateurs Goupil 3PC fonctionnant sous MS-DOS. Le support physique des données est constitué par des disquettes 5 1/4 pouces d'une capacité mémoire de 360 K.octets. Le logiciel de gestion utilisé appartient à la bibliothèque de programmes de calculs statistiques développés en Sbasic par Noiroto, Déjardin, Mullon et Savy (Publication ORSTOM, à paraître).

La base de données a pour objectifs de :

- constituer un répertoire du matériel végétal disponible,
- permettre la gestion des caféiers plantés en relation avec leur origine géographique,
- standardiser les descripteurs utilisés dans les études génétiques et agronomiques,
- faciliter l'accès et l'exploitation des données de l'évaluation.

Après avoir expliqué le codage des informations stockées, nous présenterons l'organisation générale de la base de données en dégageant les applications pratiques. Enfin, nous exposerons les problèmes rencontrés et les perspectives pour améliorer les performances du système.

1. LE CODAGE DES DONNEES

Les données codées concernent la numérotation des génotypes, leur origine géographique, leur plantation en collection et divers résultats de leur évaluation.

1.1. La numérotation des génotypes

Chaque génotype est repéré par :

- son niveau de ploïdie (format = 1 caractère),
- un identificateur de l'espèce (format = 4),
- un numéro d'ordre défini par identificateur d'espèce (format = 4).

Par exemple, l'individu 4.0001.1253 est un caféier tétraploïde (4), de l'espèce *C.arabica* (0001), numéroté 1253.

Pour les caféiers sauvages, le niveau de ploïdie et l'identificateur d'espèce sont liés : tous les *Coffea* connus sont diploïdes à l'exception de *C.arabica* qui est tétraploïde.

Néanmoins, ce codage a l'avantage de conserver l'identificateur d'espèce et le numéro d'ordre des génotypes ayant subi une duplication chromosomique. Les couples isogéniques diploïde-tétraploïde sont ainsi facilement distingués.

Ce système de numérotation est utilisable pour les hybrides des deux premières générations. L'identificateur d'espèce devient alors un identificateur de la combinaison hybride, formé à partir des deux identificateurs parentaux. Dans le cadre de cette communication, nous nous limiterons au fonctionnement de la base de données des caféiers sauvages.

1.2. L'origine géographique

L'origine des caféiers est expliquée par :

- le pays de collecte (format = 2),
- la population dans le pays (format = 3),
- la nature du matériel prélevé (graine, plantule ou axe végétatif) (format = 1),
- et éventuellement le code de l'arbre-mère.

1.3. La plantation en collection

La plantation en collection est décrite par :

- la station de recherche (format = 1),
- la parcelle dans la station (format = 2),
- la position dans la parcelle (numéro de ligne, format = 3, et rang sur la ligne, format = 2),
- l'année de plantation (format = 2),
- la nature du matériel planté (plant issu de semis, bouture ou greffe, format = 1),

- éventuellement la duplication en collection dans la même station,
- ou dans une autre station.

La plupart des génotypes sont plantés deux fois dans la même station à l'exception des descendances issues de graines (caféiers apparentés).

1.4. Les données de l'évaluation

Les données saisies de l'évaluation sont relatives à l'intensité des floraisons, aux poids de cerises fraîches récoltées et aux caractéristiques des fruits et des grains. Les champs des floraisons et des récoltes ont pour nom la date de la notation.

Les descripteurs des fruits et des grains sont :

- le poids de 100 grains à 12% d'humidité,
- le rendement en café marchand,
- le taux de remplissage des loges des fruits (deux loges potentielles par fruit),
- le taux de grains caracolis (un grain par fruit),
- le taux de loges vides,
- la teneur en caféine,
- les différentes catégories de cerises,
- la longueur du pédoncule des fruits,
- les dimensions (longueur, largeur, épaisseur).

Les quatre premiers descripteurs sont obtenus chaque année sur tous les arbres fructifères dont la récolte est pesée individuellement. La teneur en caféine est déterminée une fois pour chaque génotype. Les autres descripteurs sont utilisés selon les besoins de l'évaluation et leur liste n'est pas close.

2. L'ORGANISATION GENERALE

La base de données est formée par deux types de fichiers répondant à des contraintes et à des objectifs différents : les fichiers des individus et les fichiers de l'évaluation.

2.1. Les fichiers des individus

Les fichiers des individus sont constitués uniquement par des variables alpha-numériques représentant les codes des génotypes, de leur origine et de leur plantation en collection. Ils sont découpés par niveau de ploïdie et par identificateur d'espèce. Puis ils sont triés sur le numéro d'ordre des caféiers. L'ensemble de ces fichiers forme un répertoire du matériel végétal planté avec mention de la provenance géographique.

Ces fichiers sont créés au fur et à mesure de la plantation en collection de nouveaux génotypes. Ils sont mis à jour au moins une fois par an. Au total, ils occupent plus d'un million et demi d'octets.

Le logiciel utilisé disposant de fonctions limitées, les principales applications pratiques se résument à :

- recenser les génotypes par espèce, pays et population,
- trouver les positions en collection des caféiers d'une origine donnée,
- établir la liste des caféiers morts à remplacer chaque année,
- reconstituer les familles pour des analyses de transmission de caractères,
- rechercher les génotypes qui étaient adultes au moment de leur collecte pour étudier l'influence des pressions de sélection dans les populations naturelles.

2.2. Les fichiers de l'évaluation

Les fichiers de l'évaluation contiennent d'une part les variables alpha-numériques du code des caféiers et de leur position en parcelle et d'autre part les variables numériques des données de l'évaluation. Pour chaque type de données (floraisons, récoltes...), des fichiers différents sont créés par station, par parcelle, par espèce et par année. L'année représente un cycle annuel de reproduction. Ainsi, les données de fructification sont enregistrées avec la même année que celles des floraisons correspondantes.

Les fichiers disponibles des floraisons, des récoltes et des caractéristiques des fruits et des grains occupent environ deux millions d'octets.

3. PROBLEMES ET PERSPECTIVES

Les problèmes rencontrés proviennent du matériel et du logiciel. Il est évident que la capacité mémoire des disquettes est insuffisante. Pour le moment, les fichiers sont répartis sur 28 disquettes. Seul l'emploi d'un disque dur évitera les nombreuses manipulations de disquettes pendant les travaux de gestion.

Le logiciel utilisé n'a pas été conçu dans le cadre d'une base de données. Les fichiers sont exploités séquentiellement et séparément. Les fonctions sont donc limitées. Par ailleurs, le logiciel réserve systématiquement 8 octets par variable. Avec un programme de création de fichier tenant compte du format des variables, il sera possible de diminuer des deux tiers la taille du répertoire.

Les meilleures perspectives d'amélioration résident dans l'emploi d'un autre micro-ordinateur (Compaq 286) avec un disque dur (20 M.octets) à Montpellier. Les fichiers seront prochainement

transformés pour être exploités avec un logiciel de gestion plus performant et plus répandu tel que dBASE III. L'ouverture simultanée de plusieurs fichiers permettra d'effectuer des opérations à la fois dans le répertoire et dans les fichiers de l'évaluation. Il sera alors possible de sélectionner les caféiers d'une espèce, provenant d'une origine choisie et possédant des caractéristiques précises de floraison, récolte, etc.. Pour cela, il faudra constituer un système relationnel. Les fichiers devront être découpés autrement car certains géotypes sont plantés plusieurs fois dans la même parcelle. Le répertoire sera sans doute éclaté en un fichier des origines, indexé sur le code des caféiers, et un fichier des plantations, indexé sur la position en collection. Les fichiers de l'évaluation seront aussi indexés sur la position en parcelle. Les analyses statistiques des données seront toujours réalisées à l'aide du logiciel N.D.M.S. après transformation des fichiers.

CONCLUSION

Les objectifs fixés sont partiellement atteints. Nous avons saisi la plupart des données accumulées depuis plus de dix ans. Le répertoire du matériel végétal vivant en collection est disponible et peut être diffusé sous forme de listing. Cependant, il est indispensable d'améliorer l'accès aux données et leur traitement. L'organisation générale de la base de données devra être modifiée pour parvenir à un système performant.

La saisie des informations continue. De nouveaux fichiers sont à l'étude pour décrire :

- les populations d'origine (coordonnées géographiques, description de la forêt, ...),
- les hybridations réalisées en station (nombre de fleurs fécondées, taux de nouaison et de fructification),
- les caractères enzymatiques révélés par électrophorèse,
- les caractères morphologiques, l'architecture et la croissance des caféiers.

Enfin, il est utile de rappeler que la base de données constituée pour les caféiers sauvages est applicable aux hybrides des deux premières générations. Seules les variables de l'origine sont modifiées. Le sélectionneur peut ainsi traiter ensemble les données des hybrides et de leurs parents.

GESTION MICROINFORMATIQUE SIMPLIFIEE DES FICHIERS D'INVENTAIRES ET LEURS ASSOCIES

Philippe Bourret
(Ichtyologie)
(UR 305)

Antenne ORSTOM
Muséum d'Histoire Naturelle
Paris

1. LES PETITS RIENS

Les taxinomistes, zoogéographes et écologistes fondent l'essentiel de leur travail sur des fichiers d'inventaires documentés dont la taille est très variable suivant qu'ils sont personnels ou collectifs, circonstanciels ou permanents. S'ils veulent simplement archiver des informations, les mettre à jour et les consulter de temps en temps, le bac à fiches reste pour la plupart de ces fichiers l'expédient le moins coûteux et le plus rapide. Mais dès que l'utilisateur envisage des traitements plus élaborés, tris statistiques, tableaux ordonnés, etc., constituer un fichier informatique, même restreint, devient à terme un gain de temps, malgré toute la rigueur et le formalisme qui forcément accompagnent cette opération.

Bien souvent l'utilisateur ORSTOM n'a accès aux Centres de calcul, et même aux minis, que pour des gros fichiers et des traitements spéciaux dont il n'a pas la maîtrise. Et seul devant son micro pas très sophistiqué (le *"bilan de l'existant"* en recensait déjà 250, il y en a maintenant au moins plus du double) il balance entre trois solutions :

- *acheter un logiciel tout fait* : mais les logiciels *"grand public"* sont soit trop simplistes pour satisfaire ses besoins particuliers, soit trop généraux pour éviter que son application ne se perde dans les méandres d'une adaptation complexe ;

- *faire programmer un logiciel sur mesure* : mais les informaticiens *"maison"* sont peu nombreux et le recours à une SSII extérieure trop cher. De toutes façons cette solution est laborieuse et même parfois décevante faute d'un dialogue de niveau ;

- *écrire son propre logiciel* : et là il recule devant l'investissement. Apprendre un langage ne demande guère plus qu'un peu de persévérance, et beaucoup ont suivi avec succès des cours de Basic, mais programmer, transformer des idées en ins-

tructions exécutables est autrement plus difficile et exigeant, mieux vaut en prendre l'exacte conscience avant de s'y engager.

2. POURQUOI POI ?

Pour traiter toutes sortes de fichiers de taille moyenne, depuis les catalogues de types d'environ 200 fiches, jusqu'à la "check-list" des poissons de Nouvelle-Calédonie qui en comporte plus de 1 600, en passant par la gestion des membres de la Société Française d'Ichtyologie, les bibliographies de thèses ou d'articles, etc., j'ai écrit en 1986 le programme *POI*, et par là voulu atteindre deux objectifs :

- éviter aux utilisateurs d'apprendre un langage ou un logiciel de gestion de fichiers. Ce qui signifie permettre à tous les membres d'une équipe travaillant sur les mêmes fichiers de les créer, les consulter, les mettre à jour, les corriger, les éditer, les transférer, etc., sans risquer d'intempestives et parfois fatales modifications de structure ou de contenu ;

- améliorer les performances du progiciel par une adaptation étroite aux besoins spécifiques (par exemple : déroulement de menus et d'explications libellés dans un vocabulaire familier, pas de codages malgré leurs indéniables avantages pour l'informaticien, linéarité des opérations les plus courantes, transparence des opérations complexes), par une programmation optimisée pour la rapidité d'exécution, et par une sécurité accrue (par exemple : validation des entrées, gestion conservatrice complète et transparente des index, sauvegardes automatiques).

Dans ce système, les utilisateurs déterminent eux-mêmes l'architecture de l'ensemble des fichiers mis en relation (par exemple : fichier d'espèces associés à un fichier de synonymes et un fichier bibliographique), et la structure de chacun de ces fichiers : ce qui s'acquiert d'expérience en suivant quelques règles de conduite très simples, et ne doit au demeurant pas faire naître d'excessives angoisses puisque les structures peuvent être modifiées à tout moment.

Le programme permet alors un très grand choix dans :

- le mode d'entrée des données, de consultation, de correction, etc. ;

- la forme des sorties, qui peuvent être directes sur diverses imprimantes ou sur écran, ou bien indirectes sur des traitements de texte et des tableurs, puisque le programme crée des fichiers "ASCII séparateur".

3. LA LANGUE DE POI

Les "moulinettes" Basic sont certes des outils pratiquement irremplaçables pour transcoder des séquences de caractères, changer des formats de fichiers, transférer des données, etc., mais le Basic interprété n'est pas bien structuré, d'écriture malcommode, et surtout, comme d'ailleurs les langages compilés tel Pascal, il rend les programmes et les fichiers étroitement interdépendants.

Les générateurs d'applications, Multigen, Magic, Yes You Can ou Dialogue II, qui réalisent le logiciel dès lors qu'on leur a précisé la structure clavier-écran, permettent d'obtenir assez rapidement des applications qui "tourment". Mais répondre aux questions posées par ces générateurs est aussi difficile qu'une programmation classique, et la longueur du logiciel final rend son exécution extrêmement lente.

Il m'a donc paru préférable de recourir à un logiciel paramétrable qui gère complètement les données sur les disques et comporte un langage procédural assez complet.

POI est un programme d'environ 380Koctets, suivant ses accessoires, écrit en dBASE II version 2.4, et disponible sous CP/M et MS-DOS, version 3.0 (en cours une version dBASE III, compilée par Nantucket, alias Clipper, pour accroître sa vitesse et délivrer des versions "run-time"). Les limites des fichiers sont les suivantes :

- nombre de champs : 32 (en entrée simultanée 23 seulement) ;

- taille maximale d'une fiche : 1 000 caractères ;

- nombre théorique de fiches : 65 536 ;

- champs numériques : entiers de 0 à 999.

Son menu initial ci-dessous donne un aperçu des principales opérations qu'il réalise, avec la meilleure adéquation sur les fichiers d'espèces et leurs associés, mais en fait aussi avec une grande efficacité sur une gamme très large de fichiers.

SORTIE (et sauvegarde)

Nouvelles ENTREES dans le fichier

RECHERCHES et CORRECTIONS

INDEXATIONS du fichier

ADDITION de taxons supérieurs

LISTES, SOUS-FICHIERS, FICHIERS TEXTES

INDEX GENERAL

Consultation/mise à jour FICHER de SYNONYMES

CHANGEMENT de fichier de travail

AJOUT d'un fichier au fichier de travail

Qu'il me soit permis de tirer de cette expérience de programmation, et de la co-direction d'un projet de gestionnaire de données bibliographiques, au moins deux observations :

- la définition initiale des spécifications d'un logiciel aboutit à une sorte de cahier des charges très approximatif, qui au gré des demandes peut se transformer rapidement en une carcasse inutile ou monstrueuse, si l'on ne veille à s'imposer de sévères limitations. La définition des besoins doit nécessairement être un processus itératif, auquel il faut donner une sortie, voire un compromis, car hélas on peut tout programmer ;

- seule une étude très approfondie des performances du progiciel, notamment aux conditions limites, révèle ses "bugs", ses réactions inattendues, parfois aléatoires, et surtout ses lenteurs. C'est le prix élevé à payer pour optimiser la vitesse de procédures complexes sur des fichiers lourds, souvent d'ailleurs au moyen de solutions informatiques franchement atypiques.

Le logiciel *POI* est à la disposition des utilisateurs potentiels : il est fourni avec une documentation très complète (soixante-dix pages) qui permet une prise en main immédiate.

HYDROM : UN LOGICIEL DE GESTION DES DONNÉES D'UN RÉSEAU D'OBSERVATION HYDROMÉTRIQUE.

Gérard Cochonneau
(Informatique-Hydrologie)
(UR 502, Cadres spatiaux de l'Indépendance Alimentaire)

EMBRAPA
Empresa Brasileira de Pesquisa Agropecuaria
SuperCenter Venâncio 2.000
70333 Brasilia, DF (Brésil)

RESUME - Utilisé par le Laboratoire d'Hydrologie de l'ORSTOM (où il a été développé) et par certains services hydrologiques de pays Africains, HYDROM est un logiciel qui permet la saisie, la gestion et l'exploitation classique des données issues d'un réseau d'observation hydrométrique.

A l'origine, l'information à gérer se compose :

- de bordereaux de hauteurs d'eau remplis par des observateurs, quotidiennement ou plusieurs fois par jour ;
- d'enregistrements graphiques de hauteurs d'eau, provenant de limnigraphes ;
- d'enregistrements sur mémoire EPROM, provenant également de limnigraphes ;
- de rapport de tournées d'inspection ou d'installation de stations d'observation ;
- de résultats de campagnes de mesures de vitesses.

En raison de cette diversité, différentes techniques (saisie manuelle, digitalisation, lecture de cartouches de mémoire, dépouillement de campagnes de mesure) sont utilisées pour introduire ces données de base dans les fichiers correspondants (identification, historique des stations, hauteurs d'eau, jaugeages).

Le logiciel permet également la gestion des fichiers des données élaborées ou calculées à partir des données de base (étalonnages, débits instantanés, débits journaliers).

Une place importante est réservée à l'édition de ces informations sous forme de tableaux ou de graphiques utilisés pour contrôler la qualité des données ou pour des publications (annuaires notamment). Enfin, HYDROM possède des utilitaires pour intégrer des fichiers de données hydrométriques déjà saisies dans un autre contexte, extraire des données sous forme de fichiers en

format ASCII, préparer un fichier utilisable par d'autres logiciels (statistiques, modèles mathématiques pluies-débits).

Deux versions parfaitement compatibles sont opérationnelles, les données pouvant transiter de l'une à l'autre :

- une version pour micro-ordinateur IBM-XT ou compatibles qui permet une exploitation complète et décentralisée des données d'un bassin, d'un fleuve, d'un service ou d'une section outre-mer ;

- une version moins riche implantée sur gros système (CNUSC) et utilisée pour gérer la banque complète du Laboratoire d'Hydrologie.

Indépendamment du volume du logiciel (60 fonctions accessibles par 13 menus), de la variété des périphériques connectés (table à digitaliser, table traçante, lecteur de mémoire EPROM), il convient de signaler un certain nombre de points décisifs lors du développement :

- la définition du logiciel par enquête auprès des futurs utilisateurs ;

- les contraintes imposées par le matériel (une première version développée sur la configuration Mini-6/CIRCE de Bondy a été remise en question par le transfert du Laboratoire d'Hydrologie et son équipement en micro-informatique) ;

- les avantages et les inconvénients du langage principal (Cobol) choisi par continuité avec la première version ;

- la nécessité d'utiliser un autre langage (Basic) pour les graphiques ;

- l'accès aux fonctions du logiciel par des menus et la saisie des données et des requêtes à l'aide de grilles de saisie aussi bien sur micro-ordinateur que sur gros système ;

- la décision de développer un gestionnaire d'écran (MASQUE) et certains outils de manipulation de répertoires ;

- l'implantation du logiciel inachevé en trois lieux différents (dont une configuration en réseau de micro-ordinateurs), afin de le tester en vraie grandeur avant sa diffusion plus générale.

1. HISTORIQUE, OBJECTIFS

En 1967, le Bureau Central Hydrologique de l'ORSTOM, basé à Paris, a décidé d'informatiser la gestion des données issues des réseaux d'observation hydrométrique exploités par les sections hydrologiques Outre-Mer. Deux raisons ont motivé cette décision :

- le grand volume de données déjà collectées par les réseaux de plus en plus denses installés dans pratiquement tous les pays d'Afrique francophone depuis 1950 ;

- le bénéfice qui pourrait être tiré de cette gestion informatique pour exploiter ces données par calcul automatique.

Une fois définis les formats dans lesquels seraient conservées les données, le BCH s'est doté d'un atelier de saisie équipé d'une table à digitaliser et de perforatrices de cartes, support choisi pour stocker l'information. Parallèlement, en l'absence de personnel spécialisé, les hydrologues affectés ou de passage au BCH ont été formés au langage Fortran, choisi pour développer les programmes de gestion et d'exploitation sur les ordinateurs installés au CIRCE, centre de calcul du CNRS.

Au cours des années suivantes, un certain nombre de programmes ont été mis au point pour constituer, sinon un logiciel de banque de données, du moins un ensemble cohérent permettant le contrôle et l'exploitation des données saisies.

A partir du milieu des années 1970, les données ont été conservées sur des bandes magnétiques, la carte perforée restant le support primaire utilisé pour la saisie. Cette amélioration s'est faite sans modification des formats de données, en introduisant de nouveaux programmes de gestion, de mise à jour et d'extraction des données écrits pour la plupart en langage Cobol. Les moyens d'accès au centre de calcul ont eux aussi suivi l'évolution technologique, passant du triporteur à un terminal lourd avec lecteur de cartes, imprimante et lecteurs de bandes magnétiques.

Ce n'est qu'en 1981 que l'installation d'un mini-ordinateur au centre ORSTOM de Bondy a permis d'envisager une modernisation de la saisie des données et de l'accès au centre de Calcul et par conséquent à la banque de données. Plutôt qu'une simple adaptation de l'existant, et compte tenu de l'expérience acquise pendant les dix ans d'exploitation du système antérieur, une refonte complète a été décidée, entraînant la création de nouveaux fichiers de données jusqu'alors non informatisées, la modification du contenu des fichiers déjà gérés et, parfois, la définition de nouveaux algorithmes. Le nouveau système ainsi défini, nommé Hydrom, a été développé et utilisé partiellement jusqu'à sa presque conclusion à la fin de 1984 quand le Laboratoire d'Hydrologie s'est transféré à Montpellier.

Plusieurs faits concomitants ont alors conduit à réexaminer les orientations prises précédemment. Tout d'abord, le changement d'équipements informatiques disponibles : si l'accès à un centre de calcul (CNUSC) restait possible et s'améliorait puisque réalisé à travers des terminaux interactifs et non plus en "batch", le Mini-6 était remplacé par des micro-ordinateurs IBM-PC qui allaient ensuite être connectés en réseau. D'autre part, la mise à disposition des sections Outre-Mer de micro-ordinateurs permettait d'envisager une saisie et une exploitation locales des données hydrométriques collectées, décentralisation souhaitée aussi bien

par le Laboratoire surchargé de données à saisir que par les hydrologues expatriés, soucieux d'étudier leurs données le plus rapidement possible et in situ. Enfin, le transfert de plus en plus fréquent depuis les années 1970, de la responsabilité de l'exploitation des réseaux d'observation des sections ORSTOM aux services hydrologiques nationaux prouvait, comme le point précédent, la nécessité d'un outil de gestion décentralisable.

Une version d'Hydrom pour Goupil-III sous système d'exploitation Flex-9 a été développée à partir du milieu de l'année 1984 par Ph. Vauchel. Elle n'a pas été menée à son terme, faute de temps, et a perdu de son intérêt dès l'arrivée de l'IBM-PC. Dans sa version incomplète, elle a cependant rendu des services aux sections équipées de Goupil-III, en attendant la disponibilité de la version IBM-PC.

Diffusé en 1986 dans sa version actuelle, Hydrom a été développé au Laboratoire d'Hydrologie de Montpellier pour répondre à deux objectifs :

- le gestion et l'exploitation de la banque complète des données hydrométriques collectées par l'ORSTOM depuis l'origine des réseaux d'observation ;
- la gestion et l'exploitation décentralisées d'une partie de ces données, observées récemment sur place ou extraites de la banque complète.

Il existe donc une version du logiciel opérationnelle sur un gros centre de calcul (CNUSC) et une version opérationnelle sur les micro-ordinateurs de la famille IBM-PC, chacune d'elles intégrant des procédures d'échange de données avec l'autre. Ces deux versions sont parfaitement compatibles, en dépit des matériels et des outils différents utilisés et leur partie commune est vue par l'utilisateur de façon identique : mêmes menus, mêmes grilles de saisie, mêmes sorties imprimantes.

La figure 1 montre la succession dans le temps des versions successives, sans distinguer la période de développement de la période d'utilisation. Les flèches simples montrent que la version IBM-PC a été développée en profitant des programmes de la version Mini-6/CIRCE et que la version CNUSC a été une adaptation de la version IBM-PC. Les flèches doubles schématisent la compatibilité des données gérées par les différentes versions.

2. DESCRIPTION DES DONNÉES

2.1. Généralités sur l'hydrométrie

La finalité principale d'un réseau d'observation hydrométrique est de connaître, de manière continue, le débit des cours d'eau. Pour cela, l'hydrologue choisit certains endroits qu'il juge

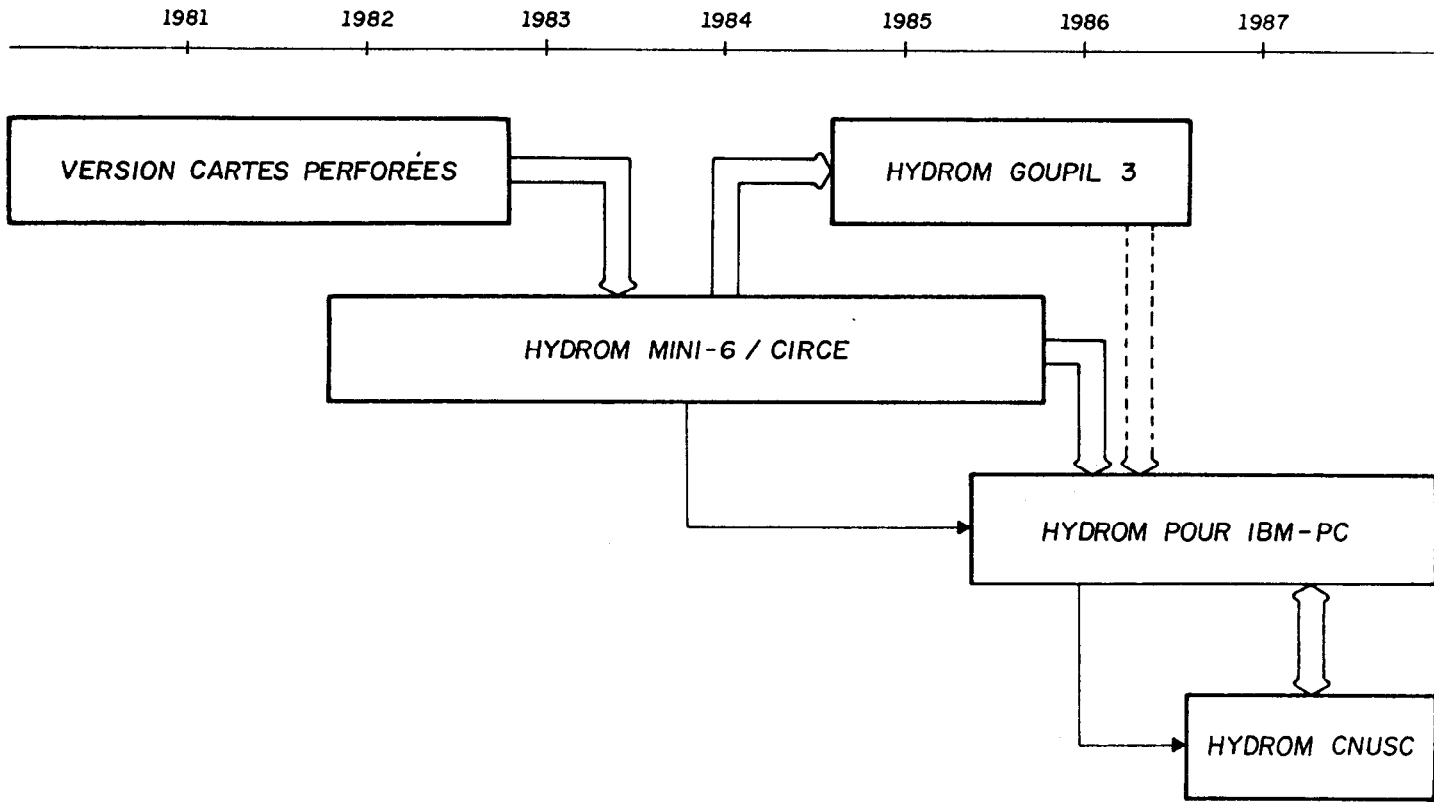


Figure 1 - Chronologie des différentes versions du logiciel.

intéressants par leur situation pour y installer une station de mesure qui contrôle les écoulements d'un bassin versant. Comme il n'est pas envisageable de mesurer directement les débits (méthodes coûteuses et peu adaptées aux conditions naturelles), on est conduit à mesurer continuellement les cotes du niveau d'eau (à l'aide de capteurs : batterie d'échelles, capteur de pression, flotteur, etc...) et à les traduire en débits grâce à une courbe d'étalonnage de la station qui fournit la relation entre la cote et le débit. Cette relation est établie expérimentalement à la suite de mesures (de vitesse en général) qui permettent de connaître le débit correspondant à la cote du niveau d'eau lors de la mesure. Ces mesures, appelées jaugeages, doivent être réalisées en différentes saisons, pour obtenir des couples cotes-débits bien répartis sur la plage de variation du niveau du cours d'eau, et de façon régulière pour accompagner l'instabilité de la relation hauteur-débit qui peut varier au cours du temps, à la suite notamment de modifications du lit de la rivière. Chaque modification importante impose de définir une nouvelle courbe d'étalonnage.

Une fois la station de mesure installée, son exploitation classique comprend donc une collecte continue des mesures de la cote des niveaux d'eau par un observateur ou un système enregistreur, des campagnes de jaugeages et des tournées d'inspection et d'entretien de périodicité variable. La partie supérieure de la figure 2 montre, de façon schématisée, les différentes étapes de cette exploitation et les informations produites par chacune d'elles.

2.2. Description des informations de base

Nous allons classer ces informations en six catégories en fonction de leur nature et du support sur lequel elles se présentent avant traitement. Nous distinguons ainsi :

- les informations signalétiques connues dès l'installation de la station (situation, équipement), les divers incidents constatés et les travaux effectués lors des tournées d'inspection ; ces données sont collectées en format libre sous forme de rapports de tournées voire même relatées par les hydrologues dans leur rapport d'activité mensuel, comme ce fut le cas à une certaine époque ;

- les fiches de jaugeages qui nécessitent d'être dépouillées avant de produire les informations utiles à l'hydrologue ; à titre d'exemple, dans le cas des jaugeages au moulinet, les données collectées (coordonnées des points de mesure dans la section de la rivière, nombre de tours effectués par l'hélice pendant un temps déterminé) sont ensuite intégrées sur toute la surface pour donner le débit total, la vitesse moyenne, la section mouillée ;

- les bordereaux mensuels de cotes, remplis par un observateur une ou plusieurs fois par jour à heures fixes ; les

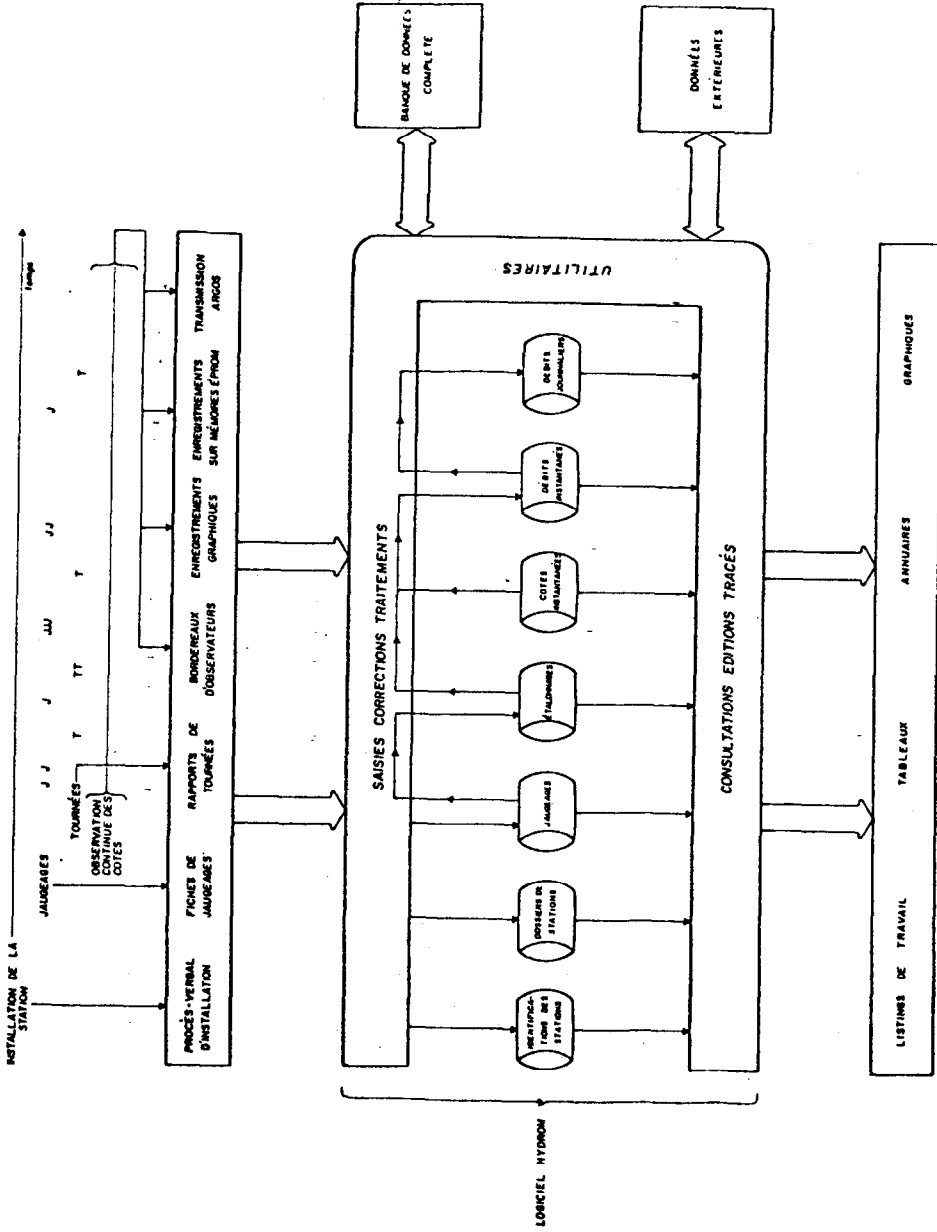


Fig. 2 - Interactions schématisées entre le logiciel et les informations administrées

bordereaux de débits, plus rares, obtenus à une usine hydro-électrique par exemple ;

- les enregistrements graphiques continus de cotes, collectés selon une périodicité variable (journalière, hebdomadaire, mensuelle) selon le régime du cours d'eau, la précision de l'information souhaitée ou l'accessibilité de la station ;

- les cotes enregistrées sur cartouches de mémoire EPROM par les limnigraphes équipés de centrale CHLOE ;

- les cotes télétransmises par les stations équipées de balises ARGOS et reçues en temps réel via une station de réception.

3. DESCRIPTION DU SYSTEME

Le but n'étant pas d'entrer ici dans une description détaillée du logiciel qui énumérerait toutes les fonctions disponibles, nous allons, dans ce chapitre, donner un aperçu de l'organisation des données, des services disponibles classés en quatre catégories (saisie contrôlée, traitements, sorties, utilitaires) et de la documentation. Le logiciel est décrit plus en détail par ailleurs, notamment dans la note qui est diffusée à titre de présentation.

3.1. Organisation des données

Les données décrites ci-dessus sont organisées en sept fichiers principaux (voir partie centrale de la figure 2) :

- identification des stations,
- jaugeages,
- étalonnages,
- dossiers de stations,
- cotes instantanées,
- débits instantanés,
- débits journaliers.

Chacun de ces fichiers fait l'objet d'un sous-système du logiciel accessible à l'utilisateur à partir d'un menu principal. Un huitième sous-système rassemble des procédures utilitaires d'extraction, d'échange de données. Ces sous-systèmes ne sont pas d'égale importance, certaines des fonctions qu'ils offrent atteignent elles-mêmes la taille d'un sous-système : les dépouillements des enregistrements de cotes, graphiques ou sur mémoire EPROM, sont parmi celles-là.

Deux fichiers supplémentaires, que nous qualifierons de secondaires, sont éventuellement nécessaires en fonction des équipements utilisés : il s'agit des fichiers des caractéristiques des hélices utilisées pour jauger et des caractéristiques de l'appareillage utilisé pour enregistrer les cotes.

Tous ces fichiers sont organisés de façon séquentielle-indexée. Les données qu'ils contiennent sont, pour la plupart, codées en binaire et rassemblées en enregistrements logiques de longueur fixe ou variable, selon la nature des données.

L'organisation physique a été pensée de manière à rendre aussi souple que possible l'utilisation du système sur des configurations aussi différentes qu'un micro-ordinateur isolé ou un réseau de micro-ordinateurs utilisés par un service décentralisé ou encore le centre de calcul CNUSC à Montpellier. Pour chaque implantation du système, les fichiers énumérés ci-dessus sont divisés en deux classes :

- les fichiers communs à l'implantation : identification des stations, hélices, appareillage des limnigraphes ; ils contiennent des informations qui intéressent tous les utilisateurs de la banque ;
- les fichiers propres à un utilisateur : tous les autres types de fichiers constitués uniquement des données propres à une étude, à une station, à une rivière, à un bassin, à une année d'un groupe de stations, etc..

Prenons pour exemple l'organisation que nous conseillons à un service hydrologique équipé d'un micro-ordinateur :

- un répertoire R0 où se trouve le logiciel ;
- un répertoire R1 des fichiers d'intérêt commun ;
- un répertoire R2 où se trouvent les autres fichiers de la banque des données hydrométriques complètes du réseau placé sous la responsabilité du service ;
- un répertoire R3 qui contient les données collectées pour l'année en cours.

Le répertoire R2 n'est donc pratiquement accédé qu'en consultation alors que les mises à jour, les critiques et les corrections des données récentes se font dans les fichiers du répertoire R3, peu volumineux donc plus aisés à manipuler et à sauvegarder après chaque altération importante. A la fin de l'année, quand toutes les données sont en place, elles sont intégrées à la banque complète. Sur le centre de calcul, la terminologie est différente mais l'organisation reste la même.

Dès l'entrée dans le logiciel, l'utilisateur doit indiquer l'étude sur laquelle il va travailler, ce qui suffira à identifier le répertoire où se trouvent les fichiers qu'il va utiliser, et dispose ensuite des fonctions offertes dans le menu des utilitaires pour extraire des données de la banque générale ou les échanger d'une étude à une autre, ou même d'une implantation à une autre par l'intermédiaire de disquettes ou par transmission de fichiers dans le cas d'un micro-ordinateur relié au CNUSC.

3.2. Saisie contrôlée

Tous les fichiers et les tables sont gérés par une fonction interactive de saisie, correction, suppression à l'aide de grilles d'écran. Un maximum de contrôles, faisant parfois intervenir d'autres fichiers (identification des stations), sont effectués pour interdire l'intrusion de données erronées dans les fichiers. Cette fonction permet également de consulter les informations du fichier.

L'utilisation de touches de fonction programmées (duplication de champs notamment) est généralisée pour aider l'opérateur dans ses tâches de saisie. La signification des touches autorisées est documentée en permanence sur l'écran quand cela est possible ou peut défiler sur la dernière ligne de l'écran, à la demande de l'utilisateur.

Pour introduire des données dans certains fichiers, il existe, en plus de la saisie/correction directe, d'autres fonctions plus complexes :

- la numérisation des enregistrements graphiques de cotes sur une table à digitaliser connectée par liaison série ;
- le transfert du contenu des cartouches de cotes par un lecteur de mémoires EPROM, connecté par liaison série ;
- le transfert des cotes télétransmises de la station de réception, toujours par liaison série ;
- le dépouillement des fiches de jaugeages.

3.3. Traitements

Certains fichiers sont aussi alimentés par calcul automatique à partir des données d'autres fichiers : les débits instantanés sont obtenus par traduction automatique des cotes à l'aide des étalonnages, les débits journaliers sont calculés à partir des débits instantanés. Tous ces traitements sont requis par l'utilisateur à travers des grilles d'écran qui lui permettent de sélectionner les stations et les périodes à considérer et réalisés immédiatement après, toujours en mode conversationnel pour informer l'opérateur du déroulement du traitement. Un compte rendu final est imprimé qui relate les opérations effectuées et les éventuelles anomalies rencontrées.

3.4. Sorties imprimées, graphiques

Pour toutes les informations gérées par le logiciel et pour les résultats obtenus à la suite de traitements classiques, des procédures d'impression sont prévues, en général sous plusieurs formes : inventaires, listings de travail, listings destinés à des publications. D'autre part, l'accent a été mis sur l'importance des graphiques, aussi bien pour contrôler la qualité des données saisies ou les comparer entre elles que pour des publications. Comme pour tout le

reste, les impressions et les graphiques sont requis en utilisant des masques d'écran.

Les graphiques sont visualisés par défaut sur l'écran, l'échelle peut être redéfinie et l'utilisateur peut profiter d'effets de zoom en sélectionnant une partie du graphique à l'aide d'un curseur mobile, pour la représenter à plus grande échelle. A tout moment, au cours de ces manipulations, le graphique affiché sur l'écran peut être obtenu, par une simple touche de fonction, sur imprimante ou sur table traçante.

3.5. Utilitaires

Nous avons déjà vu leur importance pour échanger les données entre différentes configurations qui utilisent le logiciel Hydrom. Il convient d'y ajouter la possibilité qu'ils offrent de recevoir et fournir des données dans un format séquentiel en clair, permettant ainsi d'envisager d'une part, de gérer avec le logiciel des données déjà saisies dans un autre format, d'autre part, de fournir à un utilisateur, sur support magnétique, des données extraites de la banque Hydrom dans un format facilement lisible. D'autres fonctions d'extraction plus spécifiques produisent des fichiers qui peuvent alimenter les modèles pluies-débits développés par l'UR 504 et un logiciel d'ajustement à des lois statistiques.

3.6. Fichier "Journal"

Chaque opération réalisée est datée et consignée dans un fichier qui peut être imprimé à la fin de chaque session puis détruit. Une bonne gestion du document imprimé peut permettre de contrôler les altérations faites dans les fichiers et notamment de reconstituer un fichier, en récupérant une sauvegarde antérieure et en refaisant toutes les mises à jour mémorisées depuis la date de la sauvegarde.

3.7. Documentation

La documentation diffusée se divise en plusieurs fascicules qui visent à intéresser des personnes différentes :

- une présentation du logiciel qui sera prochainement remplacée par une plaquette plus "commerciale" ;
- un manuel de l'utilisateur, qui fait la liaison entre l'hydrométrie proprement dite et le logiciel ; illustré par des exemples, il est indispensable pour préparer la mise en oeuvre des différentes fonctions ;
- un manuel de l'opérateur, qui fournit notamment toutes les informations utiles lors de la mise en oeuvre du logiciel sur les terminaux et les micro-ordinateurs (messages d'erreurs possibles, dispositions à prendre, etc.) ;

- un manuel d'aide au développement d'applications particulières : il contient, sous forme d'exemples commentés écrits en quatre langages (Basic, Cobol, Fortran et Turbo-Pascal), des routines d'accès aux fichiers de données extraits d'Hydrom par l'intermédiaire des procédures utilitaires ; ce manuel est destiné aux utilisateurs désireux de développer leurs propres applications en utilisant les données gérées par Hydrom et sans risque d'altérer les fichiers originaux.

4. DIFFUSION DU LOGICIEL

La version d'Hydrom pour micro-ordinateurs est diffusée de plusieurs manières. La plus classique consiste à fournir le logiciel et sa documentation aux hydrologues de l'ORSTOM qui en font la demande et qui ont accès à un matériel compatible. En complément, le ou les hydrologues concernés profitent parfois de leur passage à Montpellier pour se familiariser avec l'utilisation du logiciel, au cours d'un stage pratique d'une ou deux journées.

Dans le cas de services hydrologiques nationaux, la politique du Laboratoire d'Hydrologie est de céder gratuitement l'utilisation du logiciel et les fichiers de données déjà collectées aux stations qui relèvent maintenant de ces services. En contrepartie, un accord est passé afin que, chaque année, les fichiers de données nouvelles soient communiqués à l'ORSTOM et alimentent la banque de données générale. Ceci se fait souvent dans le cadre d'un accord plus vaste qui englobe d'autres activités, comme ce fut le cas pour la convention avec la République du Bénin (qui comprenait la remise en état du réseau d'observation) ou pour le projet "*Onchocercose*" de l'OMS.

Enfin, un troisième cas, plus rare, consiste à céder les programmes sources à une entité extérieure à l'ORSTOM en vue d'une adaptation du logiciel à un cas particulier. Cette cession est gratuite moyennant l'assurance :

- que les adaptations à réaliser soient mineures et ne remettent pas en cause les standards définis (formats de fichiers notamment) ;
- que la nouvelle version ainsi produite ne soit pas utilisée à des fins commerciales ;
- que l'ORSTOM soit cité aussi souvent que nécessaire comme auteur du logiciel.

Deux opérations de ce type sont en voie de réalisation ou de négociation. La première, avec le Ministère Français de l'Environnement, pour la partie du logiciel relative aux dépouillement des enregistrements graphiques et à la gestion des cotes instantanées ; des modifications étant nécessaires pour adapter le

logiciel à la numérotation des stations d'observation en vigueur en France. La seconde, avec l'EMBRAPA (Entreprise Brésilienne de Recherche Agro-Pastorale), l'adaptation principale consistant dans ce cas à une traduction du logiciel en portugais.

5. QUELQUES ÉLÉMENTS IMPORTANTS AU COURS DU DÉVELOPPEMENT

5.1. Conception du système

Une première consultation a été faite fin 1981 auprès de tous les hydrologues ORSTOM affectés ou de passage en France, afin de recueillir en vrac toutes les suggestions sur le contenu des fichiers et les fonctions que devrait être capable de réaliser le logiciel. Ce document a servi de base à l'élaboration par l'analyste d'une première proposition, discutée et affinée lors d'une réunion générale. Le développement a alors commencé et s'est poursuivi pendant une période d'environ un an au terme de laquelle une nouvelle consultation a eu lieu afin de présenter les résultats obtenus, examiner quelques points particuliers et surtout parvenir à un consensus sur certains problèmes mis en évidence par les développements déjà réalisés.

L'expérience et la culture informatique accumulées par les hydrologues lors de l'utilisation du système antérieur et le fait que la personne chargée de l'analyse et de la réalisation possédait déjà une expérience dans le domaine des banques de données hydrologiques ont grandement facilité le dialogue toujours difficile entre les utilisateurs finaux et l'analyste.

5.2. Influence de l'équipement informatique

Le type de la configuration destinée à recevoir le logiciel opérationnel a eu une influence prépondérante sur la mise au point du logiciel au niveau de la définition du système, mais aussi au niveau des outils utilisés pour le réaliser.

5.2.1. Influence sur la définition

Le système dans sa version Mini-6/CIRCE était organisé de la façon suivante :

- gestion des données sur le CIRCE ;
- saisie des données et interrogation sur le Mini-6 ;
- transfert vers le CIRCE des données saisies pour mises à jour et sollicitations à réaliser en traitements par lots ;
- saisie et corrections programmées dans un esprit d'atelier de saisie : c'est à dire saisie de tout un lot de données par un opérateur non habilité à corriger des données douteuses, puis

retour à l'utilisateur qui prépare un lot de corrections qui seront ensuite effectuées par l'opérateur.

Tous ces points ont du être réexaminés à l'arrivée des micro-ordinateurs pour lesquels :

- le système devait exister en version complète et autonome ;

- l'interactivité des procédures de saisie et correction devait être augmentée pour correspondre à une utilisation différente : saisie et correction réalisées au coup par coup par l'hydrologue lui-même.

Beaucoup de programmes et de grilles d'écran ont ainsi du être profondément modifiés pour s'intégrer à cette nouvelle ambiance. Avec le recul, on se rend compte de l'économie qui aurait été réalisée si, dès 1981, le choix avait été fait de concevoir une version du système complète et autonome sur le Mini-6 et une version sur le CIRCE. Il est regrettable que ce bon choix n'ait pas été fait, mais pouvait-on prévoir, à l'époque, l'essor que prendrait la micro-informatique dans les années suivantes ?

5.22. Influence sur le choix des langages et des outils logiciels

Lors de la définition de la première version, le choix du Cobol comme langage de programmation principal était évident, compte tenu de la bonne aptitude de ce langage à la gestion de données, partie du logiciel prépondérante comparée à la partie de calculs. L'homogénéité des compilateurs sur gros système permettant la communication entre divers langages, certains programmes appelaient des routines écrites en Fortran pour effectuer les calculs les plus complexes ou accéder à une bibliothèque graphique.

Lors du changement d'équipement provoqué par le transfert à Montpellier, deux alternatives se présentaient : continuer en Cobol ou choisir un langage plus répandu sur micro-ordinateur, comme le Basic ou le Pascal. Mises à part les qualités intrinsèques de chaque langage et ses performances sur un micro-ordinateur, un nouvel élément est apparu : comment adapter et ainsi viabiliser, le plus vite possible, un logiciel que l'on pouvait qualifier de mort-né, c'est à dire périmé par un changement d'équipements dès sa phase de conclusion, comme l'est devenue la version Mini-6/CIRCE à la fin 1984.

Tableau 1. Principaux langages et outils utilisés par la version actuelle d'Hydrom.

	Micro-ordinateur	Centre de calcul
gestionnaire de grilles d'écran	Masque	ISPF
langage de gestion de données	Level-II Cobol	Cobol
organisation de fichiers	Séquentiel Indexé	VSAM (KSDS)
langage pour les applications graphiques	Quick-Basic	---
langage pour certains calculs	Quick-Basic	Fortran
transfert de fichiers	Kermit IDEA3278	

Nous allons expliquer certains des choix résumés dans le tableau 1. Premier point important : la disponibilité d'un logiciel de grilles d'écran. En l'absence d'un tel outil adéquat utilisable en Level-II-Cobol, il a été choisi d'en développer un, appelé Masque (qui fait l'objet d'une autre communication de ce séminaire), conçu de façon à ressembler le plus possible dans sa syntaxe et surtout dans sa *"philosophie"* à celui utilisé précédemment, toujours dans le but d'accélérer l'adaptation sur micro-ordinateur.

Par son haut niveau de définition intégrant la plupart des extensions apportées à la norme Ansi-74 (notamment la gestion des fichiers séquentiels indexés), le Level-II-Cobol satisfaisait deux conditions importantes de compatibilité :

- compatibilité avec la version antérieure, permettant une adaptation sans traduction dans un autre langage des programmes déjà écrits ;
- compatibilité avec la version à développer sur le centre de calcul.

Il a donc été choisi pour toute la partie de gestion des données.

Cette dernière caractéristique de compatibilité a été un facteur favorable primordial lors de la mise au point de la version d'Hydrom sur le CNUSC. Les programmes écrits en Level-II-Cobol sont en effet pratiquement directement compilables sur le centre de calcul ; la seule adaptation nécessaire étant de substituer Masque par un autre logiciel de grilles d'écran, en l'occurrence ISPF. Le passage des fichiers séquentiel-indexés sur micro aux fichiers VSAM sur le CNUSC fut de plus totalement transparent, l'organisation séquentielle-indexée étant partie intégrante de la syntaxe du Cobol.

Par contre, la totale inadéquation du Cobol pour calculer ou pour les applications graphiques interactives envisagées sur micro-ordinateur, ajoutée à la non communicabilité entre les divers langages de programmation disponibles pour la famille IBM-PC a posé de réels problèmes. Ceux-ci ont été résolus en utilisant le langage Basic compilé et en développant des procédures d'accès aux fichiers séquentiels-indexés gérés en Cobol qui sont incompatibles avec les fichiers du même type reconnus par le Basic.

5.3. Tests en vraie grandeur

Si la première version Mini-6/CIRCE ou la version CNUSC sont conçues dans le cadre d'une implantation unique du logiciel, il n'en est pas de même de la version pour IBM-PC susceptible d'être implantée loin du lieu de développement et sur des matériels différents, compatibles (mais jusqu'à quel point ?). En plus du Laboratoire d'Hydrologie où les tests étaient faits sur micro-ordinateurs isolés ou connectés en réseau, deux sections outre-mer (Adiopodoumé et Madagascar) ont accepté d'utiliser Hydrom dès le printemps 1986, dans une version incomplète.

Tous ces tests utilisant une grande variété de données ont permis d'éliminer des erreurs parfois dépendantes des données utilisées et de mettre au point la technique à employer pour diffuser le logiciel (notice d'installation, manuel de l'utilisateur, manuel de l'opérateur).

CONCLUSIONS

Dans ce chapitre, nous allons énumérer un certain nombre de conclusions issues de la mise au point d'HYDROM. Sans vouloir leur donner valeur de lois, il faut cependant reconnaître que le volume et la diversité des fonctions développées, les moyens utilisés, les difficultés rencontrées font que ces conclusions reposent sur une expérience assez complète. Il ne faut pas perdre de vue également que ces conclusions représentent seulement le point de vue de la personne qui a développé le logiciel.

1. Equipe de développement

Il faudrait éviter aussi souvent que possible de confier la réalisation d'un logiciel aussi volumineux à une seule personne. Outre le fait que plusieurs personnes ont évidemment plus d'idées qu'une seule, ce qui suscite une émulation bénéfique, une plus grande rapidité de mise au point aurait permis de conclure plus tôt les différentes versions du logiciel et, par conséquent, de rendre son cycle de vie plus en rapport avec l'évolution des équipements. La répartition des tâches entre les différents membres de l'équipe est plus délicate à planifier car elle dépend beaucoup des capacités et des aspirations de chacun et du fait que la phase génératrice de retards est la phase de programmation.

2. Methodologie

Sur ce point, la première impression qui transparait de l'expérience exposée précédemment semble indiquer qu'il est très possible de réaliser un logiciel sans méthodologie. En fait, le développement a été partagé en plusieurs phases :

- une phase de conception réalisée avec les futurs utilisateurs ;
- une phase d'analyse pour définir les différentes fonctions du logiciel et le contenu des fichiers,
- une phase de définition de programmes, de programmation et de tests.

La nécessité du découpage en plusieurs étapes est donc naturelle et n'a pas été mieux mise en évidence (éclatement de la troisième phase identifiée) dans notre cas particulier, à cause du fait qu'une seule personne travaillait sur le logiciel. Le plus grave est en réalité que cette méthodologie "*naturelle*" n'ait pas produit une documentation correcte qui faciliterait aujourd'hui la maintenance du logiciel ou les développements ultérieurs. Là encore, la constitution d'une équipe de développement aurait certainement catalysé la réalisation de cette documentation en la rendant nécessaire à la bonne compréhension du système par ses différents membres.

3. Convivialité

La qualité de l'interface avec l'utilisateur apparait comme un élément important du "*succès*" d'un logiciel, que ce soit la facilité d'utilisation, la présentation ou la documentation en ligne ou sur papier. Cette caractéristique a été considérablement renforcée par l'usage des micro-ordinateurs et des logiciels du commerce, ce qui fait que les utilisateurs sont de plus en plus exigeants. Sur ce point, l'effort qui a été réalisé dans HYDROM a été payant et a contribué à la mise en valeur du logiciel.

4. Modularité

Dans le cas d'un système de grande envergure dont on sait que la mise au point complète sera longue, il est important d'identifier des modules plus ou moins indépendants (sous-systèmes), de les réaliser complètement et de les diffuser, même de façon restreinte. Cette mise à disposition d'une version incomplète, outre qu'elle rend crédible l'équipe de développement, peut aussi être enrichissante si les remarques justifiées de l'utilisateur sont prises en compte dans les développements ultérieurs.

5. Choix des outils logiciels

Le choix des outils utilisés pour le développement peut avoir une grande influence sur la durée de la mise au point ou même la durée de vie du système. Ceci est encore renforcé quand plusieurs versions du logiciel sont prévues sur des matériels différents. Avant de définir les programmes, et parfois même avant de réaliser l'analyse si l'utilisation d'outils logiciels est envisagée lors de cette étape, il est indispensable d'étudier les outils disponibles (langages, gestionnaires de fichiers, logiciels de grilles d'écran) et d'imaginer les avantages qu'ils pourront apporter. Plusieurs facteurs sont à considérer : les performances lors de l'utilisation finale mais aussi la portabilité, les avantages apportés lors du développement, l'utilisation répandue ou non de ces outils à l'ORSTOM ou dans la structure d'accueil.

REFERENCES BIBLIOGRAPHIQUES

BOYER (J.F.), COCHONNEAU (G.), DIEULIN (C.) et ROUCHE (N.), 1986 - *HYDROM, Manuel de l'opérateur.*

COCHONNEAU (G.), 1986 - *HYDROM, Manuel de l'utilisateur.*

COCHONNEAU (G.), 1985 - *HYDROM, Présentation.*

JACCON (G.) et SÉCHET (P.), 1980 - *Base de données hydroclimatologiques du nord-est brésilien.* Paris, ORSTOM, Série Hydrologie, vol XVII, 3-4.

ROCHE (M.), 1968 - *Traitement automatique des données hydrométriques et des données pluviométriques au Service Hydrologique de l'ORSTOM.* Paris, ORSTOM, Série Hydrologie, vol V, 3.

VAUCHEL (Ph.), 1985 - *HYDROM pour Goupil 3, Manuel d'utilisation.*

CODE FLORISTIQUE DES PTERIDOPHYTES DE GUYANE FRANCAISE

Cremers G. (1), Feuillet C. (1), Granville J.J. de (1),
Hoff M. (1) & Brisse H. (2)

(1) Centre ORSTOM de Cayenne
BP 165

97323 CAYENNE Cédex (Guyane Française)
(2) Laboratoire de Morphologie expérimentale
Institut de Botanique
28 rue Goethe
67083 STRASBOURG Cédex (France)

RESUME - Présentation d'une liste numérotée et hiérarchisée des Pteridophytes de Guyane française. Ce code floristique a pour objectif de constituer une banque de données informatisées sur les fougères à l'Herbier du Centre ORSTOM de Cayenne (CAY).

ABSTRACT - *A numbered and hierarchic list of the Pteridophytes of French Guiana. This floristic code has as its objective to establish a data bank of ferns at the Herbarium of the ORSTOM Center in Cayenne (CAY).*

AVERTISSEMENT

L'herbier du Centre ORSTOM de Cayenne (CAY) contient actuellement plus de 3.000 échantillons de Pteridophytes, représentant environ 300 espèces ; il a paru utile d'établir un inventaire des espèces de Guyane française à l'aide de la bibliographie et des échantillons récoltés au cours de ces 15 dernières années par les botanistes de l'ORSTOM ou les botanistes de passage en Guyane.

Le code floristique des Pteridophytes de Guyane est une première liste de ces taxons. En l'absence d'une source bibliographique unique et complète, ce code a rassemblé en tout premier lieu les taxons cités dans l'ouvrage du Professeur K.U. Kramer de Zürich (8) sur le Surinam ; depuis de nombreuses années, il a bien voulu nous déterminer un certain nombre de spécimens, qu'il trouve ici nos sincères remerciements.

L'inventaire bibliographique (5 à 15) s'est aussi fait principalement à l'aide des ouvrages de Flora Brasiliensis de Martius (9 & 11), de Posthumus (12), de Lemee (10), de Vareschi (15),

cependant la nomenclature reste basée sur la Flore du Surinam de Kramer (7 & 8), seules les Sélaginelles sont revues à partir du travail d'Alston, Jermy et Rankin (5).

Un certain nombre de taxons du Surinam et éventuellement du Guyana ont été rajoutés afin de rendre ce code plus général et de préparer le code floristique des Ptéridophytes des Guyanes. Au total, 1775 taxons ont été introduits, représentant 353 espèces.

L'adjonction de nouvelles espèces, corrections diverses, synonymies, etc., ayant été prévue dès le départ, il est possible de présenter périodiquement des formulaires de plus en plus complets. A chaque étape, une version sur bande magnétique est disponible pour les utilisateurs. La méthode de constitution du code (1 et 2) permet à tout moment de le faire évoluer, au fur et à mesure des nouvelles publications, puis de constituer une base de données botaniques pour la Guyane. A partir du formulaire, il est possible de coder les échantillons d'herbier des Ptéridophytes de Cayenne (CAY), de tracer l'aire de répartition des espèces et d'en préciser l'habitat.

1. INTERET D'UN FICHIER INFORMATIQUE DE LA FLORE

Le botaniste note habituellement ses récoltes sur un carnet. Il doit ensuite réexaminer ses observations pour les vérifier et les rendre comparables, au prix d'un important travail de report et de correction, avant de pouvoir en faire la synthèse. Or, aujourd'hui, l'informatique permet d'une part de réduire ces tâches répétitives et d'autre part d'améliorer les synthèses en faisant porter des comparaisons systématiques sur des ensembles d'observations plus vastes qu'avec des procédés manuels.

L'informatique implique cependant, comme investissement préalable, la constitution du fichier des taxons étudiés. Ceci est rendu possible, en floristique, par l'existence de catalogues ou de Flores de référence, comportant déjà la plupart des taxons à étudier, même si des adaptations de nomenclature et des compléments s'avèrent souhaitables.

2. CONTENU DU CODE FLORISTIQUE

Pour avoir accès à un code de ce type, le botaniste en utilise une image, matérialisée par un formulaire. Par commodité, les taxons y sont rangés, au sein de chacune des grandes divisions de la systématique, dans l'ordre alphabétique des familles, genres, espèces et taxons infraspécifiques. L'ordre alphabétique est adopté

pour faciliter la recherche d'un nom, mais il est limité au cadre de la famille, afin de laisser ensemble les genres qui se ressemblent. L'ordinateur affecte des numéros séquentiels aux noms des taxons et les imprime à leur gauche. Dans le formulaire, les noms sont tronqués à 60 caractères lorsqu'ils excèdent cette longueur ; mais, bien entendu, dans le fichier comme dans les listes récapitulatives, les noms sont écrits en entier et comprennent les noms d'auteurs. Des tableaux de correspondance entre genres, familles et grandes divisions de la systématique permettent de retrouver la place des genres dans le formulaire.

Ce code a été constitué à partir de deux fichiers. Un premier fichier associe les genres de référence aux familles selon la classification de Tryon (14). Un second fichier associe les espèces et les taxons infraspécifiques aux genres de référence. Si l'on désire un code des Pteridophytes de Guyane ordonné selon une autre classification des familles, ou uniquement classé par ordre alphabétique des genres, il suffit de modifier le premier fichier familles - genres selon la classification souhaitée, ou de la supprimer avant le traitement informatique. Ainsi un code floristique des Pteridophytes de Guyane classé par ordre alphabétique des genres est également disponible sur disquettes.

Des genres figurent dans la liste, auxquels ne semblent correspondre aucune espèce. Ce fait peut s'expliquer par deux raisons :

- soit le genre est effectivement présent en Guyane, mais aucune espèce n'a été citée ;
- soit le genre existe dans telle ou telle région de l'Amérique du Sud et sa présence est probable dans les Guyanes.

3. EMPLOI DU CODE

Un formulaire est consacré pour enregistrer chaque groupe d'observations floristiques, que ce soit sur le terrain, dans l'herbier ou à partir de listes déjà publiées. En entourant d'un trait le nom de chaque taxon observé celui-ci reste lisible. Il est préférable de relier les formulaires en fascicules afin de permettre la notation et le codage d'une série d'observations. Un formulaire, une fois rempli, est un document facile à relire et à archiver et il est utilisable directement comme un bordereau pour l'enregistrement informatique. Ainsi, seuls les numéros de code des taxons sont enregistrés les uns à la suite des autres. Dans l'ordinateur, des programmes de gestion associent les données codées aux noms latins qui leur correspondent, afin de les imprimer en clair, lors des contrôles et récapitulatifs, sous forme de listes, de comptages ou, mieux encore, de cartons géographiques localisant toutes les récoltes, plantée par

plante. Le floriste peut ainsi mobiliser les données à sa convenance, soit pour analyser un point particulier, soit pour préparer les traitements statistiques ultérieurs.

Dans la pratique, deux types de formulaires sont constitués. La figure 1 présente le premier type. Celui-ci concerne tous les taxons de référence et tous les synonymes. Ceux-ci se repèrent dans le formulaire par un nombre situé à gauche du nom, (entre parenthèses), qui indique le numéro du nom de référence. Le second type ne comporte que les taxons de référence. Il est moins volumineux et se prête mieux à la notation des observations de terrain. Dans la figure 2, les taxons sont rangés par ordre alphabétique des genres.

4. APPLICATIONS DU CODE FLORISTIQUE

Ce formulaire est utile principalement pour constituer des bases de données en botanique (3 et 4) et pour des recherches floristiques, phytosociologiques ou écologiques, il a aussi un intérêt pédagogique par sa présentation condensée et hiérarchisée des noms, depuis les familles jusqu'aux taxons infraspécifiques. Pour une étude ou une application concernant une partie déterminée de la flore, il est possible de constituer un formulaire partiel, plus maniable, ne comprenant que les taxons à étudier (par exemple, ceux qui se rencontrent dans un type de formation, comme les forêts, ou ceux qui se trouvent dans une région). Pour cela, on coche sur un formulaire complet tous les taxons retenus, puis on enregistre simplement leurs numéros et on fait ensuite imprimer le formulaire correspondant.

5. REFERENCES BIBLIOGRAPHIQUES

5.1. La fabrication d'un tel formulaire est présentée dans :

(1) BRISSE H. et GRANDJOUAN G., 1979. *Utilisation de l'ordinateur pour la constitution d'un code floristique (Exemple des Bryophytes de France)*. Actes du 7ème Colloque Informatique et Biosphère. "Flore, végétation et informatique", 71-85.

(2) BADRE F., BRISSE H., DENELLE N., FLORENCE J., GRANDJOUAN G., HOFF M. et DE RUFFRAY P., 1985. *Proposition pour un code floristique de l'Europe. 1.- Principes, constitution et utilisation, à partir d'un exemple: les Ptéridophytes d'Europe*, 77 pages. I.D.T. ORSTOM n°65, PARIS.

5.2. Exemples d'utilisation de tels formulaires :

(3) BRISSE H. - 1982. *Un exemple d'utilisation d'une banque de données phytosociologiques*. Colloque sur les Methodes ma-

thématiques appliquées à la géographie. Journées des 7-8 octobre 1982, Fac. des Lettres et Sciences Humaines - Besançon.

(4) HOFF M. & BRISSÉ H. - 1985. *Mise en place d'une base de données de Botanique tropicale. 1. Informatisation des échantillons d'herbier.* 48. p. I.D.T. ORSTOM n°65, PARIS.

5.3. Liste des formulaires constitués selon la même méthodologie :

- Formulaire floristique de Polynésie Française (J.FLORENCE) ;
- Formulaire floristique du Vanuatu (P. CABALION) ;
- Formulaire floristique de Wallis et Futuna (M. HOFF) ;
- Formulaire floristique du Togo (J.-F. BRUNEL) ;
- Formulaire floristique du Gabon (J. FLORENCE) ;
- Formulaire floristique de Guyane française (Pteridophytes, Monocotylédones et Dicotylédones (G. CREMERS et al.).

5.4. Principales références concernant les Pteridophytes des Guyanes utilisées pour constituer le formulaire :

- (5) ALSTON A.H.G., JERMY A.C. & RANKIN J.M. - 1981. *The genus Selaginella in tropical South-America.* Bull. Mus. natn. Hist. (Bot.), 9, 4 : 233-330.
- (6) HENNIPMAN E.- 1977. *A monograph of the fern genus Bolbitis (Lomariopsidaceae).* Leiden Univ. Press, 87 fig. 1 - 331.
- (7) KRAMER K.U. - 1957. *A revision of the genus Lindsaea in the New World with notes on allied genera.* Acta Bot. Néerl., Vol.6 : 97 - 290.
- (8) KRAMER K.U. - 1978. *The Pteridophytes of Suriname.* Utrecht, Flora of Suriname, 3 : 1-198.
- (9) KUHN M. - 1884. Isoetaceae, Marsiliaceae, Salviniaceae in Martius, *Flora brasiliensis.* Vol. 1, Tom. 2 : 645 - 660.
- (10) LEMÉE A. - 1955. *Flore de la Guyane française.* 1. Ed. P. Lechevalier, Paris.
- (11) MARTIUS C. - 1840-1884 - *Flora brasiliensis.*
- (12) POSTHUMUS O. - 1928. The ferns of Suriname. In "*Flora of Suriname*" : 1-196.
- (13) STOLZE R.G. - 1974. *A taxonomic revision of the genus Cnemidaria (Cyatheaceae).* Fieldiana, Bot. Vol. 37 : 1-98.
- (14) TRYON R.M. & A.F. - 1982. *Ferns and allied plants, with special reference to tropical America.* Springer Verlag, New York. 1 - 857.
- (15) VARESCHI V. - 1968. Helechos. In "*Flora de Venezuela*", 1, (1 & 2) : 1-1033.

ANNEXE : Récapitulation des Ptéridophytes de Guyane

		Familles	Genres	Espèces	Subsp.	Var.	Total
Total	Réf.	23	142	353	8	49	575
	Syn.		350	1016	2	73	1441

Total général : 2016

Le fichier floristique des Ptéridophytes de Guyane (figure 1) contient 575 taxons de références et 1441 taxons synonymes, soit un total de 2016 taxons. Un formulaire de terrain, ne comportant que les noms de référence (figure 2), extrait du fichier complet, contient 23 familles, 142 genres, 353 espèces, 8 sous-espèces, 49 variétés, soit 500 lignes disposées sur 2 pages au lieu de 10 pages pour le formulaire complet.

Numérotation des taxons

Les Ptéridophytes sont numérotées de 10000 à 12016. Dans la figure 1 ci-après, le premier numéro, à gauche, est le numéro de code. Le numéro à droite, entre parenthèses, indique, lorsqu'il existe, que le nom du taxon n'est plus valide et donc qu'il s'agit d'un synonyme. Le numéro de code du taxon valide est indiqué entre parenthèses. Dans la figure 2, les taxons sont rangés par ordre alphabétique des genres et des espèces.

**Figure 1 :Extrait du formulaire floristique des Ptéridophytes
de Guyane française contenant les taxons de référence et les
synonymes**

10005	<i>Asplenium</i> Linnaeus
10006	<i>Asplenium abscissum</i> Willdenow
10007	<i>Asplenium angustum</i> Swartz (10009)
10009	<i>Asplenium angustum</i> Swartz
10008	var. <i>appunianum</i> Baker (10009)
10010	<i>Asplenium anisophyllum</i> Auct., non Kunze (10025)
10011	<i>Asplenium anisophyllum</i> Kunze (10025)
10013	<i>Asplenium anisophyllum</i> Kunze
10012	var. <i>sanguinolentum</i> (Kunze) Hieronymus (10025)
10014	<i>Asplenium auriculatum</i> Swartz (10057)
10015	<i>Asplenium auritum</i> Swartz
10016	<i>Asplenium camptosorum</i> Mettenius (10065)
10017	<i>Asplenium cientarium</i> Swartz (10019)
10018	<i>Asplenium crenulatum</i> C. Presl (10070)
10019	<i>Asplenium cristatum</i> Lamarck
10022	<i>Asplenium cuneatum</i> Lamarck
10020	<i>Asplenium cuneatum</i> Lamarck (10022)
10021	var. <i>obtusum</i> Klotzsch (10022)
10023	<i>Asplenium dimidiatum</i> auct., non Swartz (10077)
10024	<i>Asplenium dissectum</i> Gmelin (10015)
10025	<i>Asplenium feei</i> Kunze ex Fee
10026	<i>Asplenium firmum</i> Kunze (10006)
12027	<i>Asplenium formosum</i> Willdenow
10027	<i>Asplenium haplophyllum</i> Domin
10028	<i>Asplenium hostmannii</i> Hieronymus
10029	<i>Asplenium imrayanum</i> (Hooker) Domin (10069)
10030	<i>Asplenium insigne</i> Liebmann (10065)
10031	<i>Asplenium integerrimum</i> Sprengel (10037)
10032	<i>Asplenium juglandifolium</i> Lamarck (10037)
10037	<i>Asplenium juglandifolium</i> Lamarck
10034	forma <i>brevisorum</i> Domin (10037)
10033	var. <i>eucalyptifolium</i> Domin (10037)
10035	var. <i>integerrimum</i> (Sprengel) Domin (10037)
10036	var. <i>subcrenulatum</i> Domin (10037)
10038	<i>Asplenium kapplerianum</i> Kunze (10037)
10040	<i>Asplenium laetum</i> Schkuhr (10006)
10041	<i>Asplenium laetum</i> Swartz
10039	<i>Asplenium laetum</i> auct., non Swartz (10028)
10042	<i>Asplenium loriforme</i> Hooker (10009)
10043	<i>Asplenium lunulatum</i> Swartz (10041)
12488	<i>Asplenium marinum</i> Linnaeus

Figure 2 : Extrait du formulaire floristique des Pteridophytes de Guyane française contenant uniquement les taxons de référence

10005	<i>Asplenium</i> Linnaeus
10006	<i>Asplenium abscissum</i> Willdenow
10009	<i>Asplenium angustum</i> Swartz
10013	<i>Asplenium anisophyllum</i> Kunze
10015	<i>Asplenium auritum</i> Swartz
10019	<i>Asplenium cristatum</i> Lamarck
10022	<i>Asplenium cuneatum</i> Lamarck
10025	<i>Asplenium feei</i> Kunze ex Fee
12027	<i>Asplenium formosum</i> Willdenow
10027	<i>Asplenium haplophyllum</i> Domin
10028	<i>Asplenium hostmannii</i> Hieronymus
10037	<i>Asplenium juglandifolium</i> Lamarck
10041	<i>Asplenium laetum</i> Swartz
12488	<i>Asplenium marinum</i> Linnaeus
10048	<i>Asplenium pedicularifolium</i> Saint-Hilaire
10049	<i>Asplenium perkinsii</i> Jenman
12029	<i>Asplenium pteropus</i> Kaulfuss
12038	var. <i>alleopteron</i> (Kunze ex Klotzsch) Lellinger
12037	<i>Asplenium radicans</i> Linnaeus
10057	<i>Asplenium salicifolium</i> Linnaeus
10065	<i>Asplenium serra</i> Langsdorf & Fischer
10066	var. <i>imrayanum</i> Hooker
10067	var. <i>remotum</i> Kunze
10068	var. <i>serra</i>
10069	var. <i>woodwardioides</i> (Gardner) Domin
10070	<i>Asplenium serratum</i> Linnaeus
10075	<i>Asplenium sulcatum</i> Lamarck
10077	<i>Asplenium zamiifolium</i> Willdenow
11525	<i>Azolla</i> Lamarck
11526	<i>Azolla caroliniana</i> Willdenow
10084	<i>Blechnum</i> Linnaeus
10086	<i>Blechnum asplenioides</i> Swartz
10087	<i>Blechnum brasiliense</i> Desvaux
10091	<i>Blechnum gracile</i> Kaulfuss
10094	<i>Blechnum lanceola</i> Swartz
12055	<i>Blechnum lineatum</i> (Swartz) Hieronymus
10095	<i>Blechnum occidentale</i> Linnaeus
10096	<i>Blechnum serrulatum</i> L.C. Richard
10100	<i>Blechnum volubile</i> Kaulfuss
10534	<i>Bolbitis</i> Schott
10542	<i>Bolbitis nicotianifolia</i> (Swartz) Alston

**AUBLET : LA BANQUE DE DONNÉES DE L'HERBIER
DU CENTRE ORSTOM DE CAYENNE , PREMIERS RÉ-
SULTATS**

M.Hoff, G.Cremers, C.Feuillet & J.J.De Granville

Centre ORSTOM de Cayenne
BP 165
97323 CAYENNE CEDEX (Guyane Française)

RESUME - L'herbier du Centre ORSTOM de Cayenne est en cours d'informatisation. Une première exploitation portant sur le quart des échantillons qui y sont déposés permet de mieux connaître son contenu et de présenter quelques résultats portant sur la répartition des échantillons par espèce, par collecteur par pays et par station.

SUMMARY - *The Herbarium of the ORSTOM Center of Cayenne is being computerized. A first exploitation with a fourth part of the collections deposited in Cayenne allows a better knowledge of the contents of the herbarium and gives some results about the distribution of the specimens by species, by collector, by country and by locality.*

1. INTRODUCTION

L'herbier général du Centre ORSTOM de Cayenne (CAY) comprend actuellement environ 50.000 échantillons répartis de la manière suivante :

- 40.000 échantillons de Phanérogames ;
- 2.000 échantillons de Pteridophytes ;
- 1.500 échantillons de Bryophytes ;
- 500 échantillons de Lichens ;
- 5.000 échantillons de Champignons.

Le contenu de l'herbier et sa richesse ont déjà été présentés (Cremers, 1984 et 1985). L'herbier des Phanérogames regroupe 31.000 échantillons déterminés au niveau de l'espèce et concernant 3.800 taxons, ainsi que 9.000 autres échantillons qui ne sont déterminés qu'au niveau du genre ou de la famille.

12.500 échantillons ont été enregistrés dans la banque de données, soit le 1/4 de l'herbier général. Un premier traitement informatique a été effectué sur un premier lot de 10.071 échantil-

lons de Phanérogames afin de mieux connaître la répartition des échantillons par espèce, par collecteur, par pays et par station et de tester les programmes de gestion en grandeur nature.

2. METHODE D'ETUDE

Le but de l'informatisation d'un herbier est triple. D'une part cela doit permettre une meilleure connaissance de l'herbier afin d'en faciliter la gestion (classement et rangement des échantillons, gestion des prêts et des échanges). D'autre part, cela doit faciliter le travail de détermination des échantillons par les botanistes travaillant dans l'herbier car ils ont à leur disposition la liste de la totalité des espèces et des spécimens existant dans l'herbier. Enfin, cela permet d'obtenir des résultats scientifiques tels que les listes d'échantillons par station ou par espèce afin de rédiger des flores ou des florules et de préparer les missions sur le terrain des botanistes, des écologistes ou des phytochimistes.

Les échantillons ont été enregistrés à partir des sources suivantes :

- l'herbier proprement dit (10.071 échantillons) ;
- des publications taxonomiques, révisions, flores, (550 échantillons) ;
- des publications non taxonomiques, écologique, ethnobotanique, etc. (300 échantillons, de Foresta 1981 et Riera, 1983) ;
- des listes de plantes récoltées lors des missions (700 échantillons pour la mission Smithsonian-ORSTOM-Museum du Galbao en 1986. Ces listes ont été informatisées par le Smithsonian) ;
- des carnets de récolte des collecteurs du Centre ORSTOM (1.100 échantillons) ;
- de l'inventaire des serres, du jardin botanique et de l'Arboretum (700 échantillons).

Une certaine redondance entre les informations de ces différentes sources permet certaines vérifications et mises à jour.

La méthode proprement dite consiste à saisir dans un premier temps, à partir des étiquettes des spécimens de l'herbier, les informations suivantes (Hoff et Brisse, 1985) :

- le numéro de l'échantillon, et son qualificatif éventuel (A, B, bis, ter, GY, etc.) ;
- le nom du collecteur ;
- le nom de l'espèce, ou le nom du niveau taxonomique jusqu'auquel l'échantillon a pu être déterminé (famille, genre, espèce, etc.) ;
- le nom du pays dans lequel a été effectuée la récolte ;
- le nom de la station.

Afin de faciliter la gestion et de réduire la taille des fichiers, le nom des collecteurs, des stations et des taxons sont codés lors de la saisie, à partir de formulaires de codage préétablis (Hoff et Brisse, 1986 ; Cremers et al., 1986 a, b et c).

Sur le plan de l'échantillonnage géographique et à titre d'exemple, quatre stations ont été systématiquement notées. Il s'agit de la Piste de Saint-Elie, de la station de l'Arataye, de Saül et de sa région et de la Montagne de Kaw. Elles ont été choisies en raison de l'importance des études qui y sont actuellement en cours.

L'exploitation présentée porte sur les 10.071 échantillons déjà cités, répartis entre 53 familles de Phanérogames (Figure 1, en annexe).

3. RESULTATS

Nous allons d'abord présenter quelques résultats concernant :

1. l'ensemble de l'herbier ;
2. un exemple de gestion de l'herbier d'un collecteur, l'herbier de Grenand ;
3. d'une station, la Montagne de Kaw ;
4. et d'une famille, les Commelinaceae.

Seuls de courts extraits des fichiers obtenus sont publiés ici. La totalité des données est disponible sur support informatique ou sous forme de liste imprimée.

3.1. Généralités sur l'Herbier de Cayenne

Répartition des échantillons par nom de taxon. Le premier résultat obtenu concerne la liste des espèces présentes dans l'herbier. La figure 2 montre cette liste ainsi que le nombre d'échantillons par espèce.

La figure 3 détaille la répartition des échantillons en fonction de leur niveau de détermination. On remarque sur ce tableau que, par exemple, 789 échantillons répartis entre 34 familles ne sont déterminés qu'au niveau de la famille, ce qui montre le chemin qui reste à parcourir pour une bonne connaissance de la flore guyanaise.

La figure 4 indique les espèces comportant plus de 50 échantillons dans l'herbier, les genres pour lesquels plus de 50 échantillons de l'herbier sont indéterminés au niveau de l'espèce et les familles pour lesquelles plus de 50 échantillons ne sont déterminés ni au niveau du genre ni à celui de l'espèce.

Répartition des échantillons par collecteur. Les 10.071 premiers échantillons ont été récoltés par 138 collecteurs ou groupes de collecteurs. La figure 5 présente la liste des collecteurs ayant ré-

colté plus de 100 échantillons. Dix neuf collecteurs ont rassemblés à eux seuls près de 82% des spécimens de l'herbier.

Afin de permettre plus de souplesse dans la gestion des noms de collecteurs, les collecteurs multiples sont notés indépendamment des noms simples. Par exemple : Grenand & Prévost, Prévost & Grenand sont considérés séparément de Prévost seul ou Grenand seul. Par la suite, les échantillons des deux collecteurs Grenand, Grenand & Prévost d'un côté, Prévost, Prévost & Grenand de l'autre côté sont regroupés.

Répartition des échantillons par pays. Bien que centré sur la Guyane Française, l'herbier du Centre ORSTOM s'est enrichi, surtout récemment, d'échantillons provenant du Surinam et du Guyana ceci en raison de la rédaction de "*Flora of the Guianas*". Par échange, des échantillons des autres pays d'Amérique latine ont également été intégrés à l'herbier de Cayenne. Le figure 6 montre la répartition des spécimens selon les 13 pays de provenance.

Répartition des échantillons par station de collecte. Le quart de l'herbier provient des quatre stations systématiquement prélevées. L'herbier de la Piste de Saint-Elie représente 10% de l'herbier de Guyane. Ceci est dû principalement aux inventaires écologiques qui y sont effectués. L'herbier de la région de Saül, avec près de 13% des spécimens, regroupe plusieurs localités sur une superficie nettement supérieur à celle de la Piste de Saint-Elie (Figure 7).

3.2. L'herbier GRENAND

Une exploitation détaillée de l'herbier Grenand (Ethnobotaniste ORSTOM), déposé au Centre ORSTOM de Cayenne, a été effectuée. Pour l'instant, seuls les échantillons Grenand ont été traités, à l'exclusion des spécimens Grenand & Prévost (16) et Prévost & Grenand (62). La figure 8 donne la composition de cet herbier.

La figure 9 présente la liste des numéros de collecte dans l'ordre alphabétique des noms de taxons.

La figure 10 montre la répartition des taxons dans l'ordre croissant des numéros de récolte.

3.3. L'herbier de la Montagne de Kaw

Les observations floristiques de la Montagne de Kaw ont fait récemment l'objet d'une publication (de Granville, 1986). L'inventaire des espèces végétales récoltées sur cette montagne proche de Cayenne y est donné en appendice. Cette liste a été établie à partir des carnets de récolte de de Granville, Oldeman, Cremers, Feuillet, Prévost, etc. Une comparaison avec la liste établie à partir de l'herbier montre d'une part, que quinze des

échantillons récoltés par Oldeman se trouvent à l'état d'*unicata* à Paris (P) et, d'autre part, que 17 échantillons sur 181 (soit près de 10%) ont été récoltés par des personnes extérieures à l'herbier du Centre ORSTOM de Cayenne et qui n'ont pas laissé leur carnet de récoltes.

La figure 12 fournit la liste des 12 collecteurs ayant récolté des plantes sur la Montagne de Kaw. Les collecteurs sont rangés en fonction du nombre croissant d'échantillons récoltés.

La figure 13 montre la liste des échantillons collectés (numéro de collecte et nom du collecteur) par ordre alphabétique des familles, genres et espèces.

3.4. L'herbier des Commelinaceae

Les Commelinaceae de Guyane forment une petite famille comportant 16 espèces environ réparties en une dizaine de genres.

La figure 15 donne la liste des vingt cinq collecteurs de Commelinaceae rangés en fonction du nombre croissant d'échantillons.

La figure 16 montre la liste des taxons de Commelinaceae et le nombre d'échantillons par taxon.

La figure 17 présente la liste des échantillons d'herbier rangés dans l'ordre alphabétique des taxons.

La figure 18 présente la liste des taxons de Commelinaceae rangés dans l'ordre alphabétique des collecteurs. Le chiffre figurant en dernière colonne indique le code du pays de collecte (voir plus haut).

4. CONCLUSION

Cette note indique les résultats les plus fréquemment souhaités que l'on peut obtenir par de simples programmes de gestion appliqués à un herbier informatisé. Ces résultats ont été obtenus à partir de *"dBASE II"* implanté sur un micro-ordinateur IBM-PC compatible. Les fichiers complets de cette exploitation peuvent être obtenus sur disquettes.

(Les auteurs remercient M. Henry BRISSE qui a bien voulu relire le texte et présenter des suggestions et améliorations).

REFERENCES BIBLIOGRAPHIQUES

- CREMERS G., 1984 .- *L'Herbier du Centre ORSTOM de Cayenne (CAY) a 25 ans*. Taxon, 33 (3) : 428-432.
- CREMERS G., 1985 .- *Espèces nouvelles de Guyane Française : richesse de l'herbier de Cayenne (CAY)*. Studies on the flora of the Guianas. 11, Proc., C 88 (1) : 15-37.
- CREMERS G., FEUILLET C., GRANVILLE J.J. de et HOFF M., 1986a .- *Liste informatisée des plantes de Guyane (Phanérogames et Ptéridophytes). Banque de Données AUBLET*, Herbier du Centre ORSTOM de Cayenne.
- CREMERS G., FEUILLET C., GRANVILLE J.J. de et HOFF M., 1986b .- *Liste informatisée des collecteurs de Guyane. Banque de Données AUBLET*, Herbier du Centre ORSTOM de Cayenne.
- CREMERS G., FEUILLET C., GRANVILLE J.J. de et HOFF M., 1986c .- *Liste informatisée des localités de Guyane. Banque de Données AUBLET*, Herbier du Centre ORSTOM de Cayenne.
- FORESTA H. de, 1981 .- *Premier temps de la régénération naturelle après exploitation papetière en forêt tropicale humide*. (ARBOCEL - Guyane française). Thèse de Doctorat de 3ème cycle, Ecologie générale et Appliquée, U.S.T.L., Montpellier, 32 figs., 23 Tabl. H.T., 114 p.
- GRANVILLE J.J. de, 1986 .- *Le projet de réserve biologique domaniale de Kaw. In : Le littoral guyanais. Fragilité de l'environnement*. Nature Guyanaise, Cayenne 1986, 161-178.
- HOFF M. & BRISSE H., 1985 .- *Mise en place d'une base de données en botanique tropicale. 1. Informatisation des échantillons d'herbier*. Information - Documentation Technique, ORSTOM, Paris, 65 : 1-48.
- HOFF M. & BRISSE H., 1986 .- *Constitution d'une banque de données en Botanique tropicale. La saisie des échantillons d'herbier*. Séminaire des producteurs de banques de données biologiques factuelles, Nice 27-28 fév. 1986.
- RIERA B., 1983 .- *Chablis et cicatrisation en forêt guyanaise*. Thèse de Doctorat de 3ème cycle, Université Paul-Sabatier, Toulouse, 163 p.

LISTE DES FIGURES

Figure 1 : Liste des 53 premières familles déjà enregistrées dans la banque de données

Figure 2 : Liste des espèces présentes dans l'Herbier de Cayenne

Figure 3 : Répartition des échantillons par niveau de détermination

Figure 4 : Taxons les plus représentés dans l'herbier

Figure 5 : Liste des collecteurs de l'Herbier de Cayenne

Figure 6 : Répartition des échantillons par pays de collecte

Figure 7 : Répartition des échantillons dans quatre localités

Figure 8 : Composition de l'herbier Grenand

Figure 9 : Liste des échantillons Grenand par nom de taxon

Figure 10 : Liste des échantillons Grenand par numéro de collecte

Figure 11 : Composition de l'herbier de la Montagne de Kaw

Figure 12 : Liste des collecteurs de la Montagne de Kaw

Figure 13 : Liste des échantillons de la Montagne de Kaw par nom de taxon

Figure 14 : Composition de l'herbier des Commelinaceae

Figure 15 : Liste des collecteurs de Commelinaceae

Figure 16 : Liste des noms de taxon de Commelinaceae

Figure 17 : Liste des échantillons de Commelinaceae par nom de taxon

Figure 18 : Liste des échantillons de Commelinaceae par collecteur

FIGURE 1 : Liste des 53 premières familles déjà enregistrées dans la Banque de Données

Acanthaceae	Agavaceae
Aizoaceae	Alismataceae
Amaranthaceae	Amaryllidaceae
Anacardiaceae	Annonaceae
Apocynaceae	Aquifoliaceae
Araceae	Araliaceae
Arecaceae	Aristolochiaceae
Asclepiadaceae	Asteraceae
Balanophoraceae	Basellaceae
Begoniaceae	Bignoniaceae
Bixaceae	Bombacaceae
Boraginaceae	Brassicaceae
Bromeliaceae	Burmanniaceae
Burseraceae	Butomaceae
Cabombaceae	Cactaceae
Caesalpiniaceae	Cannaceae
Capparaceae	Caprifoliaceae
Caricaceae	Caryocaraceae
Caryophyllaceae	Casuarinaceae
Cecropiaceae	Celastraceae
Characeae (Charophytes)	Chenopodiaceae
Chrysobalanaceae	Clusiaceae
Combretaceae	Commelinaceae
Connaraceae	Convolvulaceae
Crassulaceae	Cucurbitaceae
Cyclanthaceae	Mendonciaceae
Ochnaceae	

Observation : Ainsi que l'on peut s'en rendre compte, les échantillons ont été enregistrés dans l'ordre alphabétique des familles, sauf pour les Mendonciaceae qui font partie pour certains auteurs des Acanthaceae, et les Ochnaceae qui ont été saisies après le passage du spécialiste à Cayenne.

FIGURE 2 : Liste des espèces présentes dans l'Herbier de Cayenne rangées par ordre alphabétique, et nombre d'échantillons par espèce (extrait)

<i>Acanthospermum australe</i> (Loefling) O. Kuntze	3
<i>Acmella ciliata</i> (Humboldt, Bonpland & Kunth) Cassini	1
<i>Adansonia digitata</i> Linnaeus	1
<i>Adenocalymna impressum</i> (Rusby) Sandwith	1
<i>Adenocalymna inundatum</i> Martius ex de Candolle	1
<i>Adenocalymna prancei</i> A. Gentry	1
<i>Aechmea aquilega</i> (Salisbury) Grisebach	10
<i>Aechmea bromeliifolia</i> (Rudge) Baker	4
<i>Aechmea eglariana</i> L.B. Smith	1
<i>Aechmea lateralis</i> L.B. Smith	1
<i>Aechmea lingulata</i> (Linnaeus) Baker	2
<i>Aechmea melinonii</i> Hooker	5
<i>Aechmea mertensii</i> (Meyer) Schultes	11
<i>Aechmea politii</i> L.B. Smith	1
<i>Aechmea setigera</i> Martius	1
<i>Ageratum conyzoides</i> Linnaeus	15
<i>Aldina</i> sp.	1
<i>Allamanda cathartica</i> Linnaeus	26
<i>Alternanthera dentata</i> (Moench) Scheygrond	3
<i>Alternanthera flavo-grisea</i> (Urban) Urban	2
<i>Alternanthera paronychoides</i> Saint-Hilaire	5
<i>Alternanthera sessilis</i> (Linnaeus) R. Brown	1
<i>Amaranthus spinosus</i> Linnaeus	1
<i>Ambelania acida</i> Aublet	43
<i>Ambrosia artemisiifolia</i> Linnaeus	1
<i>Anacardium occidentale</i> Linnaeus	7
<i>Anacardium spruceanum</i> Benth	2
<i>Ananas annassoides</i> (Baker) L.B. Smith	4
<i>Ananas nanus</i> L.B. Smith	3
<i>Anartia meyeri</i> (G. Don) Miers	19
<i>Anaxagorea acuminata</i> (Dunal) Saint-Hilaire	4
<i>Anaxagorea brevipedicellata</i> Timmerman	6
<i>Anaxagorea dolichocarpa</i> Sprague & Sandwith	52
<i>Anaxagorea phaeocarpa</i> Martius	4
<i>Anaxagorea prinoides</i> (Dunal) Saint-Hilaire	3
<i>Anellema paraguayensis</i> C.B. Clarke	1
<i>Anemopaegma chrysoleucum</i> (Humboldt, Bonpland & Kunth)	1
<i>Anemopaegma granvillei</i> A. Gentry	1
<i>Anemopaegma ionanthum</i> A. Gentry	1
<i>Anemopaegma longidens</i> de Candolle	1
<i>Anemopaegma paraense</i> Bureau & K. Schumann	11

FIGURE 2 (suite)

<i>Anemopaegma parkeri</i> Sprague	1
<i>Anisacanthus secundus</i> Leonard	3
<i>Aniseia cernua</i> Moricand	5
<i>Aniseia martinicensis</i> (N.J. Jacquin) Choisy	2
<i>Annona ambotay</i> Aublet	11
<i>Annona crassiflora</i> Martius	1
<i>Annona densicoma</i> Martius	5
<i>Annona foetida</i> Martius	2
<i>Annona glabra</i> Linnaeus	5
<i>Annona haematanthera</i> Miquel	9
<i>Annona hypoglauca</i> Martius	3
<i>Annona montana</i> Macfadyn	7
<i>Annona muricata</i> Linnaeus f.	4
<i>Annona paludosa</i> Aublet	11
<i>Annona sericea</i> Dunal	9
<i>Annona squamosa</i> Linnaeus	2
<i>Annona tenuiflora</i> Martius	1
<i>Anredera leptostachys</i>	1
<i>Anthodiscus trifoliatum</i> G.F.W. Meyer	1
<i>Anthurium affine</i> Schott	1
<i>Anthurium atropurpureum</i> R.E. Schultes & Maguire	1
<i>Anthurium bellum</i> Schott	1
<i>Anthurium clavigerum</i> Poeppig	5
<i>Anthurium digitatum</i> (N.J. Jacquin) G. Don	2
<i>Anthurium gracile</i> (Rudge) Lindley	33
<i>Anthurium hookeri</i> Kunth	4
<i>Anthurium jenmanii</i> Engler	10
<i>Anthurium pentaphyllum</i> (Aublet) G. Don	8
<i>Anthurium polyrrhizum</i> C. Koch & Augustin	2
<i>Anthurium raimundii</i> Mayo	1
<i>Anthurium rubrinervium</i> Kunth	19
<i>Anthurium scandens</i> (Aublet) Engler	3
<i>Anthurium sinuatum</i> Bentham ex Schott	3
<i>Anthurium thrinax</i> Madison	1
<i>Anthurium trinerve</i> Miquel	28
<i>Aphelandra rubra</i> Nees	1

Observation : Le nombre moyen d'échantillons par espèce est compris entre 6 et 7. 314 espèces ne sont représentées que par un échantillon, 139 espèces par deux échantillons.

FIGURE 3 : Répartition des échantillons par niveau de détermination

Nombre d'échantillons déterminés au niveau :		Nombre de noms
de la famille	789	34
du genre	1.235	153
de l'espèce	7.760	1.249
de la sous-espèce	88	17
de la variété	190	34
de la forme	1	1
indéterminés	8	
Total	10.071	1.488

Observation : 20% de l'Herbier de Cayenne est indéterminé au niveau spécifique ou infra-spécifique.

FIGURE 4 : Taxons les plus représentés dans l'herbier*4A : Liste des espèces ayant plus de 50 échantillons*

Anaxagorea dolichocarpa Sprague & Sandwith	52
Cydista aequinoctialis (Linnaeus) Miers	53
Goupia glabra Aublet	54
Tapirira guianensis Aublet	55
Sauvagesia erecta Linnaeus	58
Dicorynia guianensis Amshoff	62
Cordia nodosa Lamarck	64
Bonafousia undulata (Vahl) de Candolle	78
Guatteria chrysopetala (Steudel) Miquel	80
Tetragastris altissima (Aublet) Martius	83
Quararibea turbinata Poiret	86

4B : Liste des genres ayant plus de 50 échantillons indéterminés au niveau de l'espèce

Cordia	52
Guatteria	58
Ouratea	64
Clusia	69
Duguetia	86

4C : Liste des familles ayant plus de 50 échantillons indéterminés au niveau du genre

Acanthaceae	51
Asteraceae	54
Burseraceae	108
Annonaceae	166

Observation : Les Annonaceae et les Burseraceae sont des familles bien représentées dans la végétation guyanaise. La détermination des spécimens est actuellement difficile.

**FIGURE 5 : Liste des collecteurs de l'Herbier de Cayenne
ayant récolté plus de 100 échantillons et nombre
d'échantillons par collecteur**

Bena (Paul)	103
Billiet (F.) & Jadin (B.)	160
Boom (B.M.) & Mori (S.A.)	326
Cremers (Georges Alexis)	653
Feuillet (Christian Patrice)	404
Foresta (Hubert de)	199
Granville (J.J. de) - B	318
Granville (J.J. de), Berg (C.C.), Jansen-Jacob & van Setten	157
Granville (Jean-Jacques de)	1237
Grenand (Pierre)	312
Jacquemin (Henri)	350
Lescure (Jean-Paul)	250
Moretti (Christian)	126
Mori (S.A.) & Boom (B.M.)	102
Oldeman (R.A.A.) - B	796
Oldeman (R.A.A.) - T	194
Oldeman (Roelof Arend Albert)	568
Prévost (Marie-Françoise)	466
Riéra (Bernard Joseph Jean Louis)	325
Sabatier (Daniel René Louis)	192
Sastre (Claude Henri Léon)	378
Sauvain (Michel)	169
Service Forestier - Guyane Française	417

Observation : Les prénoms des collecteurs ne sont écrits en totalité que lorsqu'ils sont seuls. Le B de Granville (J.J. de) - B et Oldemann (R.A.A.) - B indique qu'il s'agit des échantillons récoltés par Burgot. Ils ont leur propre numérotation. De même les échantillons Oldeman (R.A.A.) - T correspondent aux spécimens récoltés et numérotés par Tiburce. Au total 138 collecteurs ou groupes de collecteurs ont participé à la constitution de l'herbier de Cayenne.

FIGURE 6 : Répartition des échantillons par pays de collecte

Pays	Nombre d'échantillons
Guyane française	9. 570
Surinam	211
Guyana	55
Total "Flora of the Guianas"	9.836
Brésil	101
Bolivie	58
Equateur	39
Martinique	14
Paraguay	10
Mexique	8
Guadeloupe	5
Vénézuela	3
Pérou	2
Jamaïque	1
Total hors Guyane française	507
Total hors "Flora of the Guianas"	241

Observation : 95% des échantillons proviennent de Guyane et 98% de l'ensemble des trois Guyanes, domaine de la "Flora of the Guianas".

FIGURE 7 : Répartition des échantillons pour les quatre localités choisies comme exemple

Herbier de la Région de Saül	1.250 échantillons
Herbier de la Piste de Saint-Elie	944 "
Herbier de la Station de l'Arataye	450 "
Herbier de la Montagne de Kaw	181 "
Total pour les quatre localités	2.852 "

FIGURE 8 : Composition de l'herbier GRENAND

312 échantillons

242 taxons différents

Répartition des échantillons en fonction du niveau de détermination :

5	échantillons	déterminés	au	niveau	de	la	famille
27	"	"	"	au	niveau	du	genre
204	"	"	"	au	niveau	de	l'espèce
6	"	"	"	à	un	niveau	infraspécifique

Répartition des taxons en fonction du nombre d'échantillons :

185	taxons	sont	représentés	par	1	échantillon
45	"	"	"	"	2	échantillons
11	"	"	"	"	3	"
1	"	"	"	"	4	"

**FIGURE 9 : Liste des échantillons de l'herbier GRENAND
présentés dans l'ordre alphabétique des noms de taxon
(extrait)**

Acanthaceae	1621
Adenocalymna inundatum Martius ex de Candolle	
var. surinamense Bureau & K. Schumann	1455
Allamanda cathartica Linnaeus	520
	659
	1590
Alternanthera dentata (Moench) Scheygrond	2144
Ambelania acida Aublet	750
Anartia meyeri (G. Don) Miers	1923
Anaxagorea dolichocarpa Sprague & Sandwith	620
Anaxagorea phaeocarpa Martius	743
Anemopaegma paraense Bureau & K. Schumann	597
Annona	1681
	1771
	1937
Annona ambotay Aublet	601
Annona haematanthera Miquel	1400
Annona montana Macfadyen	1594
Annona paludosa Aublet	1806
Anthurium digitatum (N.J. Jacquin) G. Don	622
Araceae	778
Aristolochia dubia Hoehne	565
Arrabidaea candicans (L.C. Richard) de Candol	1241
	1831
Arrabidaea patellifera (Schlechter) Sandwith	671
Asclepiadaceae	2117
Asclepias curassavica Linnaeus	1880
Asteraceae	1540
Ayapana triplinervis (Vahl) Kahn & Robinson	1618
Bactris	1686
Bactris aubletiana Trail	1932
Bactris oligocarpa Barbosa Rodrigues	1541
Bauhinia	1683
Bauhinia guianensis Aublet	507
	1684

**FIGURE 10 : Liste des échantillons de l'herbier GRENAND
par numéro de récolte croissant (extrait)**

- 56 *Canna indica* Linnaeus
 199 *Mendoncia hoffmannseggiana* Nees
 227 *Licania heteromorpha* Bentham
 345 *Callichlamys latifolia* (Richard) K. Schumann
 376 *Pseuderanthemum* sp.
 407 *Stizophyllum riparium* (Humboldt, Bonpland & K
 449 *Combretum laxum* N.J. Jacquin
 456 *Justicia*
 459 *Hymenocallis tubiflora* Salisbury
 460 *Clusia grandiflora* Splitgerber
 473 *Vernonia remotiflora* L.C. Richard
 487 *Rollinia mucosa* (N.J. Jacquin) Baillon
 490 *Merremia macrocalyx* (Ruiz & Pavon) O'Donell
 497 *Tapirira guianensis* Aublet
 506 *Maripa scandens* Aublet
 507 *Bauhinia guianensis* Aublet
 509 *Caryocar glabrum* (Aublet) Persoon
 510 *Ouratea*
 512 *Odontadenia cururu* (Martius) K. Schumacher
 515 *Oedematopus octandrus* (Poeppig & Endlicher) P
 520 *Allamanda cathartica* Linnaeus
 521 *Martinella obovata* (Humboldt, Bonpland & Kunt
 523 *Quararibea guianensis* Aublet
 537 *Vismia cayennensis* (N.J. Jacquin) Persoon
 538 *Peltogyne pubescens* Bentham
 547 *Hirtella glandulosa* Sprengel
 548 *Licania micrantha* Miquel
 549 *Odontadenia cururu* (Martius) K. Schumacher
 554 *Guatteria discolor*
 558 *Schlegelia violacea* (Aublet) Grisebach

Observation : Les numéros manquants correspondent soit aux familles non encore enregistrées, soit à des unicum déposés dans l'Herbier Wayapi de Cayenne, soit à des échantillons déposés dans d'autres herbiers.

FIGURE 11 : Composition de l'herbier de la Montagne de Kaw

181 échantillons

113 taxons différents

Répartition des échantillons en fonction du niveau de détermination :

9 échantillons déterminés	au niveau de la famille
21 " "	au niveau du genre
81 " "	au niveau de l'espèce
2 " "	à un niveau infraspécifique

Répartition des taxons en fonction du nombre d'échantillons :

76 taxons sont représentés par	1 échantillon
24 " " "	2 échantillons
7 " " "	3 "
2 " " "	4 "
4 " " "	5 "

FIGURE 12 : Liste des collecteurs de la Montagne de Kaw et nombre d'échantillons

Blanc (Patrick)	1
Oldeman (Roelof Arend Albert)	2
Jacquemin (Henri)	3
Prévost (Marie-Françoise)	3
Skog (L.E.) & Feuillet (C.)	3
Billiet (F.) & Jadin (B.)	4
Leeuwenberg (Anthonius Josephus Maria)	5
Petrov (Ivan)	8
Oldeman (R.A.A.) - B	13
Feuillet (Christian Patrice)	24
Cremers (Georges Alexis)	38
Granville (Jean-Jacques de)	70

Observation : Les collecteurs sont rangés par nombre d'échantillons collectés croissant. Sur les 138 collecteurs de l'Herbier de Cayenne, seul 12 d'entre-eux ont réalisé des récoltes sur la Montagne de Kaw.

FIGURE 13 : Liste des échantillons d'herbier de la Montagne de Kaw par nom de taxon (extrait)

Acanthaceae

Anisacanthus secundus Leonard	
Cremers (Georges Alexis)	5734
	6912
Oldeman (R.A.A.) - B	2390
Justicia sp.	
Cremers (Georges Alexis)	5759
Ruellia rubra Aublet	
Granville (Jean-Jacques de)	2907

Annonaceae

Annonaceae sp.	
Feuillet (Christian Patrice)	2112
	2248
Granville (Jean-Jacques de)	6661
Duguetia sp.	
Granville (Jean-Jacques de)	6687
	6704
Froesiodendron surinamense (R.E. Fries) R.E. Fries	
Cremers (Georges Alexis)	6913
Guatteria chrysopetala (Steudel) Miquel	
Cremers (Georges Alexis)	4238
Oldeman (R.A.A.) - B	650
Rollinia exsucca (Dunal) A. de Candolle	
Granville (Jean-Jacques de)	2992
Unonopsis stipitata Diels	
Oldeman (R.A.A.) - B	634
Xylopia sp.	
Leeuwenberg (Anthonius Josephu)	11821

Apocynaceae

Apocynaceae sp.	
Feuillet (Christian Patrice)	176
	2838
	2855
Anartia meyeri (G. Don) Miers	
Granville (Jean-Jacques de)	2942

Observation : Les échantillons sont rangés par ordre alphabétique des familles, genres et espèces. Les collecteurs sont rangés par ordre alphabétique et par numéro de collecte croissant.

FIGURE 14 : Composition de l'herbier des Commelinaceae

87 échantillons

17 taxons différents

Répartition des échantillons en fonction du niveau de détermination :

6 échantillons déterminés	à la famille
5 " "	au genre
74 " "	à l'espèce
2 " "	au niveau infraspécifique

Répartition des taxons en fonction du nombre d'échantillons :

7 taxons sont représentés par	1 échantillon
1 taxon est représenté	" 2 échantillons
2 taxons sont représentés	" 3 "
2 " " "	" 4 "
1 taxon est représenté	" 5 "
1 " " "	" 6 "
1 " " "	" 10 "
1 " " "	" 11 "
1 " " "	" 31 "

Répartition des échantillons en fonction de leur pays d'origine :

1 Guyane	76 échantillons
3 Surinam	5 "
8 Bolivie	3 "
4 Brésil	2 "
2 Guyana	1 "

Observation : Le chiffre précédant le nom du pays est le numéro de code du pays utilisé dans la figure 18.

FIGURE 15 : Liste des collecteurs de Commelinaceae de l'Herbier de Cayenne

Black (G.A.), Vincent (C.) & Colmet Daage (F.)	1
Ducatillon (C.) & Gély (A.)	1
Foresta (Hubert de)	1
Granville (J.J. de) - B	1
Granville (J.J. de) - B-C	1
Granville (J.J. de) - T	1
Hallé (Francis)	1
Jacquemin (Henri)	1
Merlier (Henri)	1
Raynal-Rocques (Aline Marie)	1
Granville (J.J. de), Allorge (L.), Cremers, Gérts-	2
Sastre (C.) & Sastre (F.)	2
Sastre (C.), Sastre (F.) & Moretti (C.)	2
Sauvain (Michel)	2
Hassler (Emil)	3
Lescure (Jean-Paul)	3
Grenand (Pierre)	4
Prévost (M.F.) & Grenand (P.)	4
Feuillet (Christian Patrice)	5
Oldeman (R.A.A.) - T	5
Oldeman (R.A.A.) - B	6
Sastre (Claude Henri Léon)	6
Prévost (Marie-Françoise)	7
Cremers (Georges Alexis)	11
Granville (Jean-Jacques de)	15

Observation : Les collecteurs sont rangés par nombre d'échantillons collectés croissant. Granville (J.J. de) - B-C signifie qu'il s'agit des spécimens récoltés par Chantal de Granville et Burgot. Granville (J.J. de) - T signifie qu'il s'agit des échantillons récoltés par Tiburce. Ils ont leur propre numérotation indépendante de celle de de Granville J.J. Un collecteur de Cayenne sur cinq a échantillonné des Commelinaceae.

FIGURE 16 : Liste des taxons de Commelinaceae de l'Herbier de Cayenne et nombre d'échantillons par taxon

<i>Aneilema paraguayensis</i> C.B. Clarke	1
<i>Aneilema umbrosum</i> (Vahl) Kunth	
subsp. <i>ovato-oblongum</i> (Palisot de Beauvois) Morton	1
<i>Commelina obliqua</i> Vahl	1
<i>Dichorisandra</i>	1
<i>Dichorisandra aubletiana</i> Roemer & Schultes	1
<i>Floscopa glabrata</i> Hassler	
var. <i>hirsuta</i> C.B. Clarke	1
<i>Gibasis geniculata</i> (N.J. Jacquin) Rohweder	1
<i>Callisia repens</i> Linnaeus	2
<i>Commelinopsis rufipes</i> (Seubert) D.R. Hunt	3
<i>Floscopa peruviana</i> C.B. Clarke	3
<i>Commelina</i>	4
<i>Commelina benghalensis</i> Linnaeus	4
<i>Commelinopsis glabrata</i> D.R. Hunt	5
Commelinaceae	6
<i>Buforrestia candolleana</i> C.B. Clarke	11
<i>Commelina erecta</i> Linnaeus	11
<i>Dichorisandra hexandra</i> (Aublet) Standley	31

Observation : Les noms de taxon sont rangés par nombre d'échantillons croissant. La moitié des espèces de Commelinaceae ne sont représentées que par un seul échantillon.

FIGURE 17 : Liste des échantillons de Commelinaceae de l'Herbier de Cayenne par nom de taxon (extrait)

Commelinaceae

<i>Aneilema paraguayensis</i> C.B. Clarke	
Hassler (Emil)	11347
<i>Aneilema umbrosum</i> (Vahl) Kunth	
subsp. <i>ovato-oblongum</i> (Palisot de Beauvois) Morton	
Sastre (Claude Henri Léon)	1743
<i>Buforrestia candolleana</i> C.B. Clarke	
Cremers (Georges Alexis)	7663
	8307
	8603
Feuillet (Christian Patrice)	867
Granville (J.J. de), Allorge (7365
Granville (Jean-Jacques de)	3658
	3903
	5524
	6883
Oldeman (R.A.A.) - T	786
Prévost (Marie-Françoise)	1818
<i>Callisia repens</i> Linnaeus	
Cremers (Georges Alexis)	7791
Prévost (Marie-Françoise)	1327
<i>Commelina</i> sp.	
Cremers (Georges Alexis)	8668
Feuillet (Christian Patrice)	1383
Prévost (Marie-Françoise)	1590
Sauvain (Michel)	267
<i>Commelina benghalensis</i> Linnaeus	
Cremers (Georges Alexis)	8421
Lescure (Jean-Paul)	704
Raynal-Rocques (Aline Marie)	19928
Sastre (C.) & Sastre (F.)	30

Observation : Les échantillons sont rangés par ordre alphabétique des genres et espèces, par ordre alphabétique des collecteurs et par numéro de récolte croissant. Cette liste se situe habituellement à la suite de la description des espèces dans les Flores.

**FIGURE 18 : Liste des taxons de Commelinaceae de
l'Herbier de Cayenne par collecteur**

Black (G.A.), Vincent (C.) & C			
<i>Commelinopsis glabrata</i> D.R. Hunt	17559	54-	1
Cremers (Georges Alexis)			
<i>Commelinopsis glabrata</i> D.R. Hunt	6176		1
<i>Floscopa peruviana</i> C.B. Clarke	6715		1
<i>Dichorisandra hexandra</i> (Aublet) Standley	6968		1
<i>Buforrestia candolleana</i> C.B. Clarke	7663		1
<i>Callisia repens</i> Linnaeus	7791		1
<i>Commelina erecta</i> Linnaeus	8012		1
Commelinaceae sp.	8306		1
<i>Buforrestia candolleana</i> C.B. Clarke	8307		1
<i>Commelina benghalensis</i> Linnaeus	8421		1
<i>Buforrestia candolleana</i> C.B. Clarke	8603		1
<i>Commelina</i> sp.	8668		1
Ducatillon (C.) & Gély (A.)			
<i>Commelina erecta</i> Linnaeus	61		1
Feuillet (Christian Patrice)			
<i>Dichorisandra hexandra</i> (Aublet) Standley	542		1
<i>Buforrestia candolleana</i> C.B. Clarke	867		1
<i>Commelina</i> sp.			
Commelinaceae sp.	2945		1
Foresta (Hubert de)			
<i>Dichorisandra hexandra</i> (Aublet) Standley	494		1
Granville (J.J. de) - B			
<i>Dichorisandra hexandra</i> (Aublet) Standley	5202		1
Granville (J.J. de) - B-C			
<i>Dichorisandra hexandra</i> (Aublet) Standley	48		1
Granville (J.J. de) - T			
<i>Dichorisandra hexandra</i> (Aublet) Standley	1077		1
Granville (J.J. de), Allorge (
<i>Buforrestia candolleana</i> C.B. Clarke	7365		1
<i>Dichorisandra hexandra</i> (Aublet) Standley	7460		1
Granville (Jean-Jacques de)			
<i>Dichorisandra hexandra</i> (Aublet) Standley	294		1
<i>Commelinopsis glabrata</i> D.H. Hunt	1034		4
<i>Dichorisandra hexandra</i> (Aublet) Standley	2084		1
	2193		1
<i>Commelinopsis rufipes</i> (Seubert) D.R. Hunt	2299		1
	2537		1
<i>Commelina erecta</i> Linnaeus	3180		1

FIGURE 18 (suite)

Buforrestia candolleana C.B. Clarke	3658	1
	3903	1
Dichorisandra hexandra (Aublet) Standley	4347	1
	4637	1
Floscopa peruviana C.B. Clarke	4713	1
Buforrestia candolleana C.B. Clarke	5524	1
Commelinaceae sp.	6713	1
Buforrestia candolleana C.B. Clarke	6883	1

Observation : Les taxons sont rangés par ordre alphabétique des collecteurs et par numéro de collecte croissant. Le dernier chiffre à droite indique le numéro du nom du pays de collecte. Le code des numéros des pays se trouve sur la figure 14. G.A. Black associe à son numéro de collecte les deux derniers chiffres de l'année de collecte : (54-17559) signifie que l'échantillon a été récolté en 1954.

LE SYSTEME DE DONNEES CROISIERES

Jean-Jacques Lechauve
(Informatique)
(Atelier Informatique de Brest)

Antenne ORSTOM auprès de l'IFREMER
BP 337
29273 Brest Cédex (France)

RESUME - Vouloir connaître l'océan a conduit les chercheurs à organiser des campagnes dans le but de mesurer le plus grand nombre de paramètres. Au cours du temps la technologie mise en oeuvre a évolué et la masse d'informations recueillies est devenue sans cesse plus importante. L'ordinateur est dès lors l'outil indispensable à l'exploitation de cette information. Si sa fonction calcul est l'une des plus appréciées, la plus importante est sans conteste celle de gestion des données (une donnée est une information enregistrée sur un support). Développer un système de données doit répondre au double objectif suivant :

- représenter un certain "monde réel" ;
- optimiser la fonction disponibilité des données.

La représentation d'un certain "monde réel"

Dans tout processus d'analyse la première phase consiste à délimiter le domaine d'application de la réalisation. Pour confirmer cet adage nous dirons que le système "CROISIERES" a été développé dans le but d'offrir aux utilisateurs finals les produits de première nécessité pour étudier les données de type "profil vertical". On appelle profil vertical l'ensemble des valeurs d'un paramètre (température par exemple) mesurées depuis la surface de la mer jusqu'à une profondeur déterminée en un point précis de l'océan et à un instant donné.

Un dialogue fructueux avec les utilisateurs permet de définir précisément les besoins. Ainsi la structure conceptuelle de l'information fut clairement définie. Les points suivants seront abordés :

- définition des informations du système et des relations ;
- propriétés qualitatives et quantitatives des informations ;
- les besoins des utilisateurs.

La phase suivante fut celle où les soucis d'implémentation dans un système informatique furent pris en compte. Dans un projet de grande taille il est important de scinder cette phase en deux parties : la *structure d'accès* et la *structure physique*. La simplicité du projet ne nécessitant pas cette subdivision les deux structures seront exposées globalement. C'est à ce niveau que furent choisies :

- les méthodes de codage de l'information ;
- la façon de gérer la mémoire secondaire ;
- la règle du jeu pour les droits d'accès (concurrence entre plusieurs demandeurs simultanés et problèmes de mise à jour) ;
- la méthode d'accès.

La disponibilité des données

L'ampleur du phénomène micro-ordinateur et le fulgurant développement des télécommunications font que cette fonction disponibilité des données peut être assurée dans les meilleures conditions. Les projets de grande taille mettront en oeuvre des solutions centralisées au niveau calculateur et offriront des services à leurs clients via les grands réseaux de télécommunications. L'autre philosophie consiste, dans le cas de projet plus modeste, à dupliquer le système de données sur les micro-ordinateurs des clients. C'est dans le cadre de cette seconde solution que seront présentés les moyens retenus pour rendre accessible l'information.

Un système de gestion de bases de données possède en général ses propres langages de description et de manipulation des données. Dans le cadre du projet "*CROISIERES*" ces langages n'existent pas. C'est la raison pour laquelle il est préférable d'employer le terme de système de données. Par contre tout un environnement logiciel est proposé aux utilisateurs. Nous distinguerons trois classes d'outils qui s'adressent à des publics différents :

- la bibliothèque des routines d'accès. Seul l'interface avec le langage Fortran 77 sera exposé. Cependant afin de s'ouvrir à un plus vaste public de programmeurs une bibliothèque Quick-Basic est disponible ;
- les utilitaires de manipulation de données. Ces programmes interactifs sont utilisés pour introduire les données dans le système. Ils permettent également de mettre en conformité les versions précédentes des fichiers ;
- les programmes d'exploitation. Quelques modules seront présentés, le but de l'exposé étant de présenter seul le

système de données. Le schéma général de chacun de ces programmes sera exposé sans entrer dans le détail de la fonction mathématique ou graphique appliquée sur l'ensemble de données sélectionnées.

En conclusion

A l'instant où ces lignes sont écrites le système de données avec les deux premiers niveaux de l'environnement logiciel existe. Le développement de ce produit se poursuit à l'Atelier Informatique de Brest et une version définitive sera sans doute disponible vers la fin de l'année. Ce système, toujours opérationnel dans sa première version sur mini-calculateur, semble avoir fait ses preuves. L'intérêt du travail actuel réside dans le fait que le produit va devenir totalement indépendant du calculateur hôte. Pour des besoins plus intenses un portage sur une station de travail pourra être réalisé très simplement et dans des temps records.

Tant au sein du laboratoire qu'à bord d'un navire océanographique l'intérêt de ce système de données est d'offrir immédiatement et convivialement tous les services de base dont le chercheur a besoin dans l'accomplissement de sa mission.

INTRODUCTION

Vouloir décrire l'océan nécessite la mise en oeuvre de moyens technologiques plus ou moins importants. Les techniques d'acquisition utilisées profitent régulièrement des progrès réalisés notamment en électronique et en informatique. Les conséquences immédiates sont :

- une meilleure qualité de l'information ;
- un plus grand volume de données.

Le système 'CROISIERES' se propose de prendre en charge une classe particulière de données : les profils verticaux. On appelle *profil vertical* l'ensemble des valeurs d'un paramètre (température, par exemple) mesurées depuis la surface jusqu'à une profondeur donnée en un point de l'océan et à un instant déterminé. Afin de bien introduire cette classe de données il semble bon de présenter succinctement deux systèmes d'acquisition. Le premier (figure 1) est le type classique de station à bord d'un navire océanographique. Le second (figure 2) beaucoup moins coûteux peut être installé sur des unités diverses et plus particulièrement sur les navires marchands pour réaliser la surveillance de l'océan le long des lignes de navigation. Cette dernière technologie est exploitée intensivement depuis 1979 dans le programme SURTROPAC (SURveillance TRansOcéanique du PACifique).

SYSTEMES D'ACQUISITION

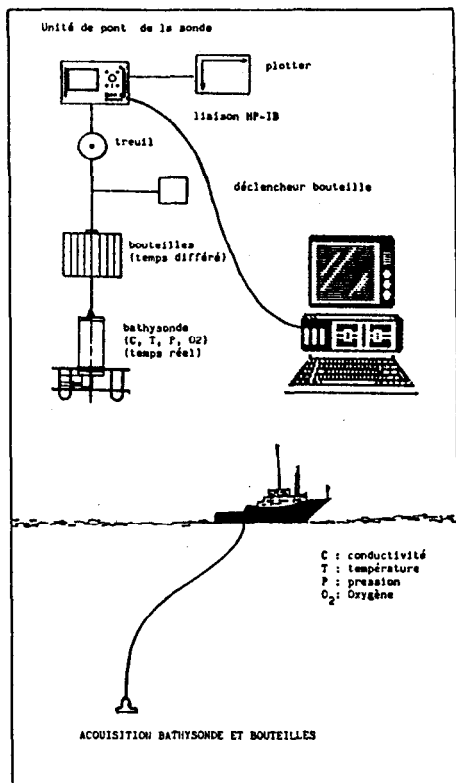


figure 1

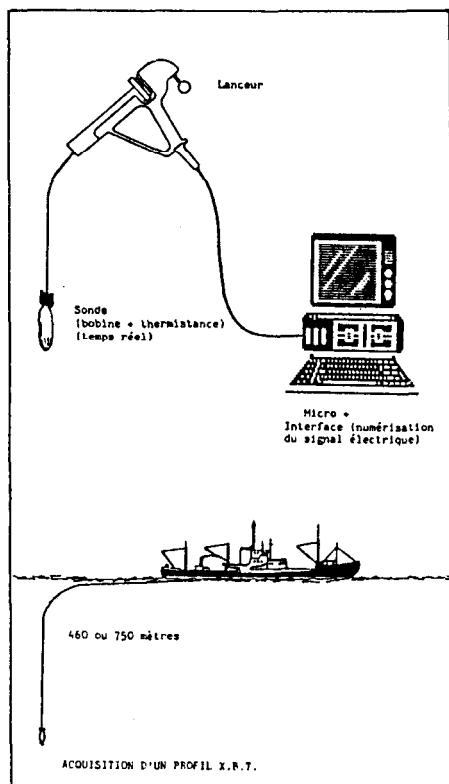


figure 2

Cette phase d'acquisition produit au terme d'une suite de traitements un lot de données (grandeurs physiques) stockées dans une mémoire de masse (cassette, disquette, etc.). C'est l'instant où le système 'CROISIERES' sera mis en oeuvre. Comme pour tout système de bases de données ses objectifs sont :

- représenter "*un certain monde réel*";
- optimiser la fonction disponibilité des données dans un système informatique.

1. REPRESENTER "UN CERTAIN MONDE REEL"

1.1. La structure conceptuelle

1.1.1. Les besoins des utilisateurs

L'ensemble des requêtes formulées par les utilisateurs peuvent se résumer à accéder à un ou plusieurs paramètres mesurés dans une zone géographique et un intervalle de temps.

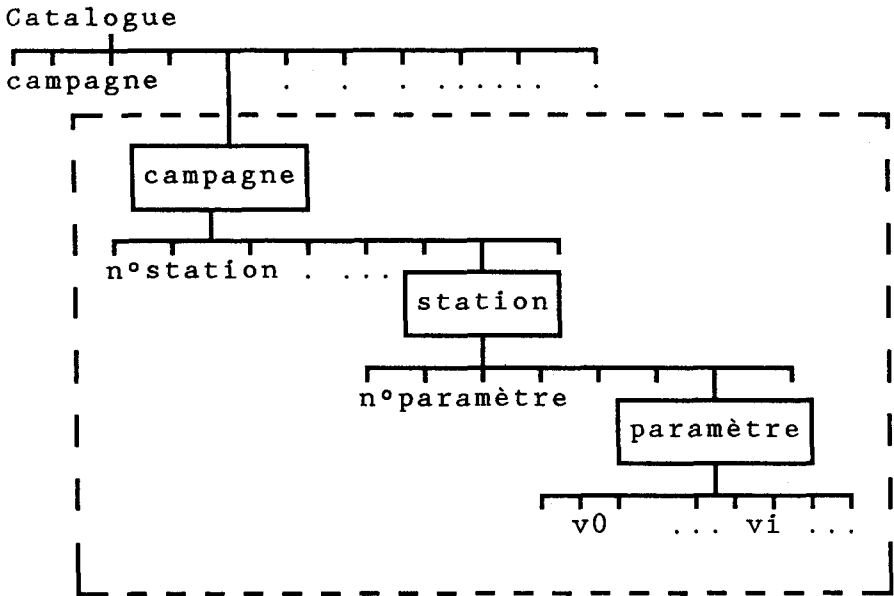
On remarque la simplicité de cette expression. On distinguera le cas particulier où les intervalles de définition des critères de sélection sont réduits à un point. Ainsi la station est définie par l'ensemble des paramètres mesurés en un point et à un instant données. On note que la technique d'acquisition est transparente dans l'expression des besoins. Cela assure que toute évolution technologique ne remettra pas en cause la structure conceptuelle qui doit jouer le rôle de constante de référence. Enfin la nature des opérateurs mathématiques ou graphiques appliqués aux données n'implique pas la formulation de contraintes quant à l'organisation des données.

1.1.2. Les informations et les relations

L'examen du problème fait apparaître les notions suivantes : campagne, station, paramètre, *intervalle de temps* et *zone géographique*.

On peut formaliser les objets et les liens qui les unissent par un schéma (figure 3).

Un rectangle en trait plein représente un groupe répétitif dont les composantes sont les éléments situés à l'extrémité des lignes droites descendantes. Une valeur terminale soulignée représente un index, c'est-à-dire que pour chaque valeur de cet élément, il n'existe qu'une et une seule occurrence du groupe répétitif.



- figure 3 -

Les notions d'intervalle de temps et de zone géographique sont introduites lors de l'emploi des procédures d'accès aux données. Ces références au temps et à l'espace sont le résultat de calculs et de tests et par conséquent n'interviennent pas encore à ce stade de l'analyse. Le catalogue assure dans l'organisation des données 'CROISIERES' un double rôle : celui de "répertoire logique" des fichiers dits 'CROISIERES' et celui de support d'une sélection spatio-temporelle. Le contenu du rectangle en tireté sur la figure précédente est l'image exacte d'un fichier 'CROISIERES'.

1.13. Les propriétés qualitatives des informations

Ce sont les propriétés qui sont vraies quelles que soient les valeurs affectées aux données :

- une station contient toujours un paramètre qui définit une relation d'ordre sur l'ensemble des valeurs des autres paramètres. C'est la pression ou l'immersion dans la plupart des cas ;
- le nombre de paramètres est différent d'une station à l'autre ;
- le nombre de niveaux varie d'une station à l'autre ;
- etc..

1.14. Les propriétés quantitatives des informations

On trouve essentiellement des contraintes de cardinalité :

- une campagne est constituée d'au plus 164 stations ;
- une station est composée d'au moins deux paramètres ;
- une station contient au plus 30 profils ;
- un profil est une suite de 1 024 valeurs au plus ;
- si la valeur d'un paramètre à un niveau donné n'est pas connue, une valeur spéciale devra lui être affectée afin que tous les profils d'une station possèdent le même nombre de composantes ;
- etc..

1.2. La structure d'accès et la structure physique

Dans le système 'CROISIERES' cinq types d'article sont formellement définis :

- 1 l'en-tête 'CROISIERES' (campagne) ;
- 4 la liste des stations d'une campagne ;
- 2 l'en-tête station ;
- 5 la liste des paramètres d'une station ;
- 3 un paramètre d'une station.

Cette notion d'article est indépendante de la façon dont l'information est codée physiquement. Elle permet uniquement de distinguer les groupes élémentaires d'information auxquels on désire accéder. Les articles de type (4) et (5) bien qu'étant des sous-articles seront traités comme des articles à part entière.

L'ensemble des informations après codage devient un ensemble de nombres entiers qui seront stockés sous forme de mots de 16 bits.

1.21. Description de l'en-tête 'CROISIERES'

Date début de la campagne

Mot 01 : Année (deux derniers chiffres)

Mot 02 : Mois

Mot 03 : Jour

Date de fin de la campagne

Mot 04 : Année

Mot 05 : Mois

Mot 06 : Jour

Positions limites des stations

Mot 07 : Latitude minimum en degrés minutes N > 0

Mot 08 : " maximum " " "

Mot 09 : Longitude minimum en degrés minutes W > 0

Mot 10 : " maximum " " "

Informations nécessaires à l'adressage

Mot 11 : Plus grand numéro de station

Mot 12 : Nombre de stations dans le fichier

Mot 13 : Nombre de paramètres différents présents dans la campagne (limité à 42)

Mot 14 : Nombre de secteurs dans le fichier .

Liste des paramètres mesurés

Mots 15 à 56 : codes des paramètres. Tous ne sont pas obligatoirement présents dans chaque station.

Les codes utilisés sont ceux de la banque hydro du BNDO (Banque Nationale des Données Océanologiques IFREMER).

1.22. Descriptif de l'en-tête station

Mot 01 : Numéro de la station

Date de la station

Mot 02 : Année (2 chiffres)

Mot 03 : Mois

Mot 04 : Jour

Mot 05 : Heure

Mot 06 : Minute

Position de la station

Mot 07 : Latitude en degrés minutes N > 0 (+ mot 61)

Mot 08 : Longitude en degrés minutes W > 0 (+ mot 62)

Mot 09 : Fond en mètres

Mot 10 : Nombre de niveaux

Mot 15 : Nombre de paramètres mesurés

Paramètres météo

Mot 11 : Température de l'air sec (dixièmes de degré)

Mot 12 : Température de l'air humide

Mot 13 : Direction du vent (dizaines de degrés)

Mot 14 : Vitesse du vent (noeuds)

Indicatif du bateau

Mots 28 à 31 C'est la seul cas où des mots contiennent des codes ASCII. La chaîne est cadrée à gauche (huit caractères max. complétés à blanc)

Mots réservés

Mots 16 à 21 PROFILER

Mots 27 à 32 XBT

Mots 33 à 60 libres

Mots 63 à 68 METEO

Informations complémentaires sur les positions

Mot 61 : Centièmes de minute latitude (pas de signe)

Mot 62 : Centièmes de minute longitude

1.23 L'adressage dans un fichier 'CROISIERES'

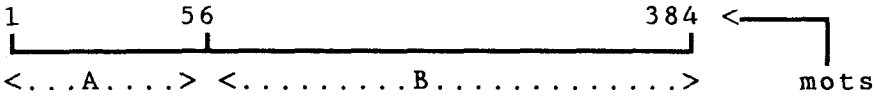
La mémoire de masse est considérée comme un ensemble de morceaux de 256 octets adressables directement. On choisit d'appeler secteur un tel morceau. Lors de la première implémentation du logiciel (sur mini-ordinateur Hewlett-Packard) cette notion de secteur était physique. Bien qu'il n'y ait plus de raison matérielle on conserve la terminologie par souci de compatibilité avec la version précédente. Une campagne est constituée d'une suite de secteurs numérotés séquentiellement à partir de 1. Au sens du système d'exploitation de l'ordinateur hôte ce sera un fichier à accès aléatoire par bloc de 256 octets. Le système de gestion de fichiers de la machine se charge de la gestion physique de la mémoire de masse.

Un fichier 'CROISIERES' contient à la fois les données physiques et toutes les informations nécessaires à leur adressage. La technique de 'navigation' dans un fichier utilise essentiellement des tables.

Descriptif des différents secteurs**Secteurs 1 à 3 :**

A : en-tête campagne (56 valeurs)

B : table des stations (2 x 164)

**Mots particuliers :**

A(12) nombre de stations dans la campagne

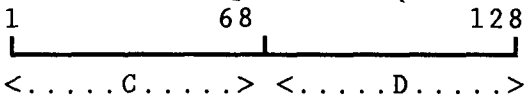
B(1,i) numéro de la i_ème station

B(2,i) adresse secteur de la i_ème station

Secteur station :

C : identification de la station (68 valeurs)

D : table des paramètres (2 x 30 valeurs)

**Mots particuliers :**

C(10) nombre de niveaux dans la station NI

C(15) nombre de paramètres NP

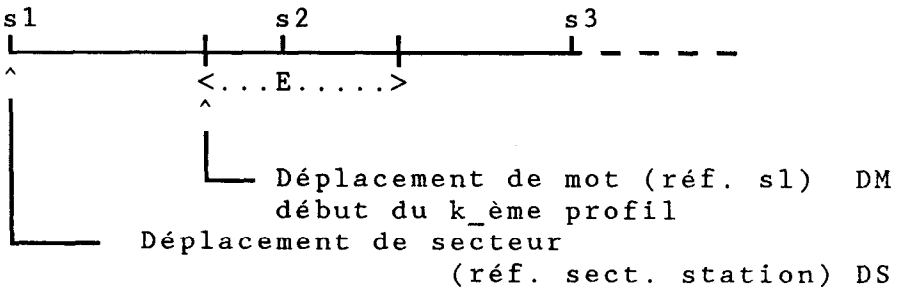
D(1,k) code du k_ème paramètre

D(2,k) adresse du k_ème paramètre DS DM

Secteur(s) paramètre :

E : valeurs du profil (NI valeurs)

sn : frontières de secteur

**Cas particulier :**

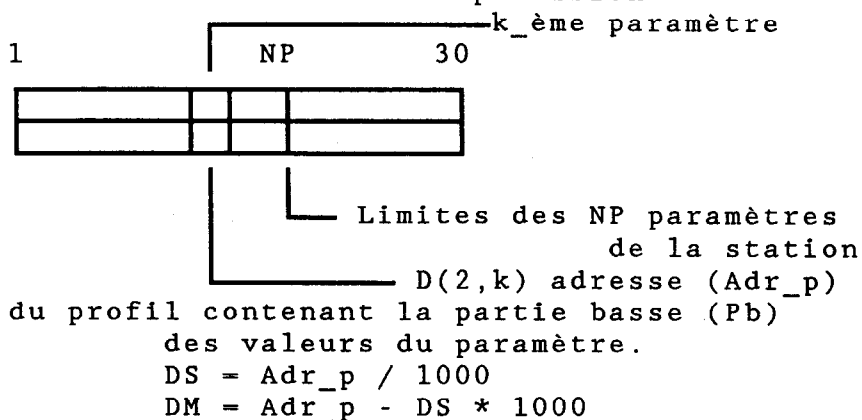
Un paramètre suivant son intervalle de définition peut nécessiter deux profils pour son stockage en mémoire de masse. Dans ce cas le mode de codage du paramètre est qualifié de grande précision.

Table des paramètres D :

bits 0 à 7 code paramètre

bits 8 à 10 code échelle

bit 14 code précision



Supposons que le k-ème paramètre soit codé sur deux profils. L'interface d'entrée / sortie va alors rechercher au-delà de la limite NP (de NP+1 à 30) un mot D(1,j) contenant le même code paramètre. Une fois trouvé l'interface accède au second profil avec l'adresse D(2,j). Ce second vecteur contient la partie haute (Ph) des valeurs du paramètre. La valeur définitive du paramètre est calculée de la manière suivante : $VP = Ph * 10000 + Pb$ (entier sur 32 bits). Cette dernière valeur est alors divisée par la puissance de 10 définie par la valeur du code échelle.

2. LA DISPONIBILITE DES DONNEES

Le second objectif auquel doit répondre une base de données est d'optimiser la fonction disponibilité des données dans un système informatique. Les choix faits pour le système 'CROISIERES' ont été influencés par l'extraordinaire souplesse d'emploi des micro-ordinateurs compatibles PC. La puissance des microprocesseurs qui les animent (INTEL 80286, par exemple) conjuguée au bon rendement des mémoires de masse donnent toute garantie de succès à la mise en place de 'CROISIERES' sur de telles configurations. L'avantage immédiat, étant donnée la popularité de ces matériels, est la possibilité pour tout demandeur de traiter dans son bureau le lot de données qui l'intéresse.

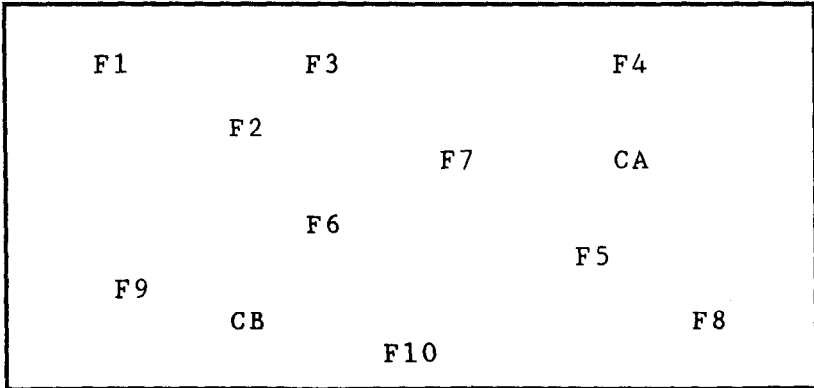
Ce système s'adresse à un public varié d'utilisateurs. Pour les initiés à la programmation BASIC ou FORTRAN une bibliothèque d'outils est proposée. Pour ceux qui ne pratiquent pas cette technique, un ensemble de programmes permet d'exploiter les

données du système. Ces modules ont été rédigés à la demande de chercheurs et leur nombre actuel fait que la majeure partie des questions posées relativement à cette classe de données se trouvent résolues. Enfin des utilitaires sont à la disposition de ceux qui veulent créer de nouveaux fichiers.

2.1. Création et mise à jour du système de fichiers

Le système 'CROISIERES' est un ensemble de fichiers au format 'CROISIERES' auquel s'ajoute éventuellement un ou plusieurs catalogues. La mise en place de cet ensemble est très simple à réaliser. Les fichiers sont créés dans un répertoire (notion DOS) soit par les outils proposés dans le logiciel, soit par simple recopie de disquettes issues d'un centre serveur (Atelier Informatique de Brest par exemple). Un catalogue est créé puis mis à jour en y consignnant l'existence des fichiers voulus. Plusieurs catalogues peuvent cohabiter en se partageant les fichiers d'un même répertoire. La figure 4 représente schématiquement cette situation.

Un répertoire DOS



Fi Fichier 'CROISIERES'

Ck Catalogue

- figure 4 -

On remarque l'existence de deux catalogues. Le premier indexé par la lettre A contient les fichiers 1,2,3,7,8,9 et le second indexé par B repère les fichiers 3,4,5,6,7,8. Chacun de ces catalogues a été défini par l'utilisateur en fonction de ses besoins.

2.11. Création d'un fichier 'CROISIERES'

Deux possibilités se présentent. La première consiste à mettre le lot de données initiales à un format reconnu par les outils du logiciel, puis d'exécuter dans l'ordre les programmes de création de fichiers. Cette solution nécessite un minimum de programmation qui se justifie par la grande variété des sources de données. La seconde possibilité consiste à mettre à jour les fichiers au format 'CROISIERES' reconnu par la version précédente du logiciel. Les fichiers à traiter peuvent être dans ce cas sur bande magnétique en archives. Décrivons succinctement ce cas de figure dans le cadre de l'Antenne de Brest. L'ordinateur serveur (accessible par réseau au standard X25) est un BULL DPS7. Un utilitaire (GFTDPS) permet de lire les fichiers 'CROISIERES' sur bande d'archives et de les transcoder sur disque à un format compatible avec les exigences du réseau. Le fichier (caractères) est ensuite transmis et capturé sur micro. Un programme opère alors le transcodage inverse donnant ainsi un fichier binaire qu'il reste à mettre en conformité avec les nouvelles spécifications de format. Un utilitaire (PATCHCRO) est proposé à cet usage.

2.12. Le catalogue et sa mise à jour

Le catalogue répond à une double préoccupation. La première est de regrouper logiquement des fichiers 'CROISIERES' en fonction de l'étude à réaliser. La seconde est de permettre la définition et la mémorisation d'une sélection.

L'utilisateur crée un catalogue avec l'utilitaire CATAL puis déclare toujours grâce à ce même programme la liste des fichiers à prendre en compte. Ce catalogue contient alors l'ensemble des données vitales à la procédure de sélection (dates, positions géographiques, paramètres, etc.). Le module CATAL permet également de supprimer des fichiers d'un catalogue, de vérifier si les fichiers déclarés sont toujours présents dans le répertoire du disque, enfin de marquer les stations qui satisfont aux définitions des critères d'une sélection. Le dernier ensemble de stations sélectionnées reste marqué tant qu'une autre sélection n'est pas définie.

2.2. Les interfaces

Une bibliothèque de routines (SPCRO.LIB) est à la disposition des utilisateurs qui veulent développer leurs propres applications. Parmi les points d'entrée quatre sous-programmes sont tout particulièrement chargés de l'accès aux données. Ces modules constituent l'interface d'entrée/sortie du système 'CROISIERES'. Cet interface a été développée en FORTRAN-77 et tout récemment en QwickBASIC. Cela offre à un plus large public de programmeurs la possibilité de développer des outils personnalisés. Décrivons succinctement ces quelques routines :

LECPA routine de lecture des cinq types d'articles définis précédemment. Ces types correspondent au cinq options d'entrée du sous-programme :

- 1 - En-tête croisière
- 2 - En-tête station
- 3 - Valeur d'un paramètre d'une station
- 4 - Liste des stations d'une campagne
- 5 - Liste des paramètres d'une station

ECRPA routine de recopie des valeurs d'un paramètre d'une station.

EFFPA routine de positionnement des "flags" :

- 1 - Protection d'une campagne en écriture
- 2 - Elimination d'une station d'une campagne
- 3 - Elimination d'un paramètre d'une station

ECRSTA routine de mise à jour de l'en-tête station. Les mots vitaux au fonctionnement des procédures d'adressage ne peuvent pas être modifiés.

Remarque : Dans un système d'exploitation qui supporte la multiprogrammation le problème du partage des fichiers doit être abordé. Dans la version mini-ordinateur le système de gestion de fichiers du calculateur hôte permettait d'ouvrir les structures de données avec des protections. Le système MS/DOS dans sa forme actuelle est mono-tâche et par conséquent aucune disposition n'est à prendre à cet égard.

2.3. Les programmes

2.3.1. Les utilitaires

Les programmes de création de fichiers évitent à l'utilisateur courant d'approfondir la structure 'CROISIERES' et lui assurent la garantie d'obtenir des fichiers conformes aux spécifications. Un minimum de programmation reste cependant à la charge de l'utilisateur pour mettre ses données au format d'entrée reconnu par les utilitaires. Les formats proposés par l'IFREMER (banque hydro) et par le SHOM (format ECHANGE pour les bathythermographes) sont également supportés par le logiciel. Un utilitaire (PATCHRO) permet la mise en conformité des fichiers générés

sous la version précédente. Ce programme offre quelques services supplémentaires. Enfin deux outils au caractère particulièrement inquisiteur permettent aux initiés de fouiller et modifier les parties les plus intimes des fichiers 'CROISIERES'.

2.32. Les programmes d'exploitation

Ce paragraphe s'éloigne du sujet de l'article proprement dit, mais il est important d'y consacrer quelques lignes. En effet pour l'utilisateur final le LOGICIEL CROISIERES c'est avant tout cet ensemble de programmes.

Chacun d'eux remplit une fonction de calcul, de visualisation, d'édition ou de gestion de données. Un programme accède aux données de deux manières différentes :

1 - accès à un catalogue (une sélection a été préalablement définie par l'emploi de CATAL option sélection) ;

2 - accès une par une aux stations d'une campagne particulière. Dans ce cas le programme est beaucoup plus interactif.

Certains programmes ne proposent que l'accès de type 1 ou celui de type 2 , par contre d'autres modules laissent le choix à l'opérateur. Faisons un rapide tour d'horizon des produits :

Edition :

- Analyse d'une campagne (diverses options)
- Edition d'une campagne " "
- etc..

Visualisation :

	Ecran	Plotter
- Visualisation et validation des profils	X	.
- Tracés divers de profils	.	X
- Relation $R(P_x, P_y)$	X	X
- Relation $R(P_x, \text{grad}(P_y))$	X	X
- Relation $R(P_x, P_{y1}/P_{y2})$	X	X
- Cartographie d'une campagne	.	X
- etc..		

Calcul :

- Calculs dynamiques
- Valeurs intégrées (contenu thermique)
- Calcul des gradients
- Détermination d'isovaleurs
- Etude du courant géostrophique
- Etude de la linéarité de certaines relations
- etc..

Gestion de données :

- Transcodages divers pour échange
- Extraction et création de fichiers pour diverses analyses statistiques (interface avec le logiciel MAGIK Méthode d'Application de la Géostatistique et de l'Interpolateur du Krigeage)
- etc..

CONCLUSION

L'implémentation dans un environnement PC du système de données qui vient d'être décrit se poursuit à l'Atelier Informatique de Brest. La part restante du travail consiste à adapter les programmes d'exploitation de la version précédente. Ce travail est particulièrement intéressant du fait de l'effort de standardisation entrepris en matière de programmation (Fortran 77 pour le langage et GKS pour l'interface graphique). Un premier volume a été rédigé (voir références bibliographiques) et deux autres sont en préparation. Le volume II présentera l'utilisation des catalogues et une partie des programmes d'exploitation. Enfin un troisième volume sera consacré aux outils graphiques du logiciel.

Ce produit 'CROISIERES' est déjà utilisé depuis presque une dizaine d'années et est appelé à servir encore les océanographes de l'ORSTOM. Cette nouvelle version s'est enrichie grâce aux remarques formulées par ses nombreux utilisateurs. C'est dans ce contexte de dialogue permanent avec les équipes de recherche que l'Informatique ORSTOM doit trouver ses sources d'inspiration, que cela conduise à développer par nos propres forces des produits ou à exploiter les logiciels du marché.

REFERENCES BIBLIOGRAPHIQUES

A. CABANES *Banques de données*. Institut d'Informatique d'Entreprise. Cours CNAM 1976.

J.R. DONGUY *Le climat dépend aussi de l'océan*. ORSTOM Actualités n°16 Mars 1987.

A. MORLIERE, J. CITEAU, J. NOEL *Recueil de notes techniques et de programmes élaborés sur une chaîne d'acquisition de données*. République de COTE D'IVOIRE. CRO. Archives scientifiques vol.2,n°3,Août 1977.

A. MORLIERE *Proposition d'équipement en matière d'hydrologie à l'ORSTOM*, 1979.

J.J. LECHAUVE *Le logiciel CROISIERES*. Document technique. Antenne ORSTOM. Brest, 1979.

J.J. LECHAUVE *Le système de gestion des données CROISIERES*.

Vol.I. Document technique n°38. Antenne ORSTOM de Brest, 1987.

GEOMER : BASE DE DONNEES DE GEOPHYSIQUE MARINE

François Missègue
(Géophysique)
(UR 106)

Centre ORSTOM de Nouméa
BP A5
Nouméa Cédex (Nouvelle Calédonie)

RESUME - GEOMER est une base de données de géophysique marine. La mise en oeuvre a été motivée par la masse de données de plus en plus importante devenue très difficile à manipuler. Le seul impératif était d'avoir un accès direct aux données par croisière, par profil ou par fenêtre géographique.

L'ossature choisie est une grille géographique sur laquelle un module de base se positionne. La donnée de base est un point de mesure. Il est lié à un module par sa position dans l'espace et à une croisière par son temps. Des réseaux de pointeurs interactifs permettent d'identifier le module récepteur et de positionner les points de mesure sur les fichiers. Quel que soit le mode d'extraction choisi, l'opération est une recherche spatiale à sélectivité variable.

La base est implantée sur HP1000 et dispose de 30 MO de mémoire de masse. L'utilisation en E/S du format international d'échange MGD77 rend GEOMER compatible avec toutes les grandes bases actuelles. Le codage binaire a permis de réduire au minimum le volume de mémoire de masse.

En développement, tous les logiciels seront traduits en Fortran 77 MicroSoft 3.31 afin d'assurer une bonne portabilité et de permettre une implantation sur compatible IBM PC-AT avec possibilité de mise en réseau local type Omninet et de branchement futur au réseau TOMPAC.

Un logiciel de création est à l'étude. Il permettra une création de base de données à la carte. La grille géographique, le type de données et les structures internes des fichiers devront être définis par l'utilisateur.

1. OBJECTIFS

La base de données GEOMER a été mise en oeuvre afin d'archiver, de conserver et de faciliter l'accès à l'importante masse de données récoltées pendant plus de dix ans par l'équipe de géologie-géophysique ORSTOM de Nouméa. Une première analyse a permis d'établir un cahier des charges en fonction, d'une part, des besoins formulés, et d'autre part des impératifs techniques.

Les impératifs techniques étaient :

- implantation sur le HP 1000 du centre ;
- allocation de 30 MO de mémoire de masse ;
- masse des données à stocker : environ 1.000.000 de points de mesure ;
- une compatibilité des formats E/S permettant des échanges de données avec les bases extérieures existantes.

Les besoins formulés par les futurs utilisateurs se résument à un accès direct aux données selon trois options :

- par fenêtre géographique ;
- par croisière ;
- par profil.

La structuration et l'organisation interne étant laissées au choix du concepteur.

2. DONNEES

Les données incorporées à GEOMER sont des données de géophysique marine. Elles sont situées dans le temps et dans l'espace. La donnée de base est le *point de mesure* qui comporte 11 champs. GEOMER met actuellement à la disposition des utilisateurs environ 350.000 mesures de bathymétrie, 200.000 mesures de gravimétrie et 350.000 mesures de magnétisme.

La cadence actuelle d'incorporation de croisières anciennes qui doivent être entièrement retraitées et des croisières nouvelles, incorporées au fur et à mesure de leur réalisation, permet d'évaluer la masse de données qui sera stockée fin 1988 à environ 1.200.000 mesures de bathymétrie, 1.200.000 mesures de magnétisme et environ 500.000 mesures de gravimétrie.

3. CONCEPTION ET FONCTIONNEMENT

La situation des points de mesure dans le temps et l'espace étant les deux seuls critères d'identification possible, le mode d'incorporation retenu devait respecter la chronologie et la position spatiale des données.

La structure de base de GEOMER est une grille géographique, qui a été volontairement limitée à la zone de travail des utilisateurs actuels. Ses limites sont 10°N et 40°S en latitude, et 90°E et 120°W en longitude (Figure 1).

Cette grille est découpée en modules appelés stations selon un pas de 1/2 degré. Une station correspond à une couverture géographique de 1/4 de degré carré (Figure 2). Le tronçon de croisière entrant dans une station est appelé traversée de station.

Des réseaux de pointeurs interactifs permettent, lors de l'incorporation, de situer tout point de mesure dans l'espace grille et de les retrouver lors des recherches pour extraction.

3.1. Les fichiers

L'organisation de GEOMER s'articule autour de six fichiers principaux : INFOG, DGNDX, DGLST, PROFIL, STATN, CROISR contenant les mises à jour des pointeurs, les réseaux de pointeurs, les informations scientifiques, techniques et administratives des croisières incorporées et les descriptifs des traversées de station et des profils. Les données équipées de leurs différents pointeurs sont stockées dans des fichiers MESxxx. Chaque croisière a son propre fichier MESxxx.

INFOG est un compteur contenant la mise à jour des pointeurs.

DGNDX est la grille géographique. Chaque enregistrement, formé d'un champ, correspond à un des 30.301 1/4 de degré carré. L'adresse de l'enregistrement est calculée à partir de la position géographique du sommet Sud-Ouest du 1/4 de degré carré ou station.

L'état du champ indique si la station a été ou non traversée par une croisière quelconque. Si la station a été traversée, il donne l'adresse à laquelle la première traversée réalisée est répertoriée dans le fichier DGLST.

DGLST contient par ordre chronologique de réalisation toutes les traversées de station. Un enregistrement correspond à une traversée ; il contient toutes les informations permettant :

- d'identifier la croisière au cours de laquelle la traversée de la station a été exécutée ;
- de situer la station sur la grille géographique ;
- d'accéder au fichier descriptif de station STATN ;

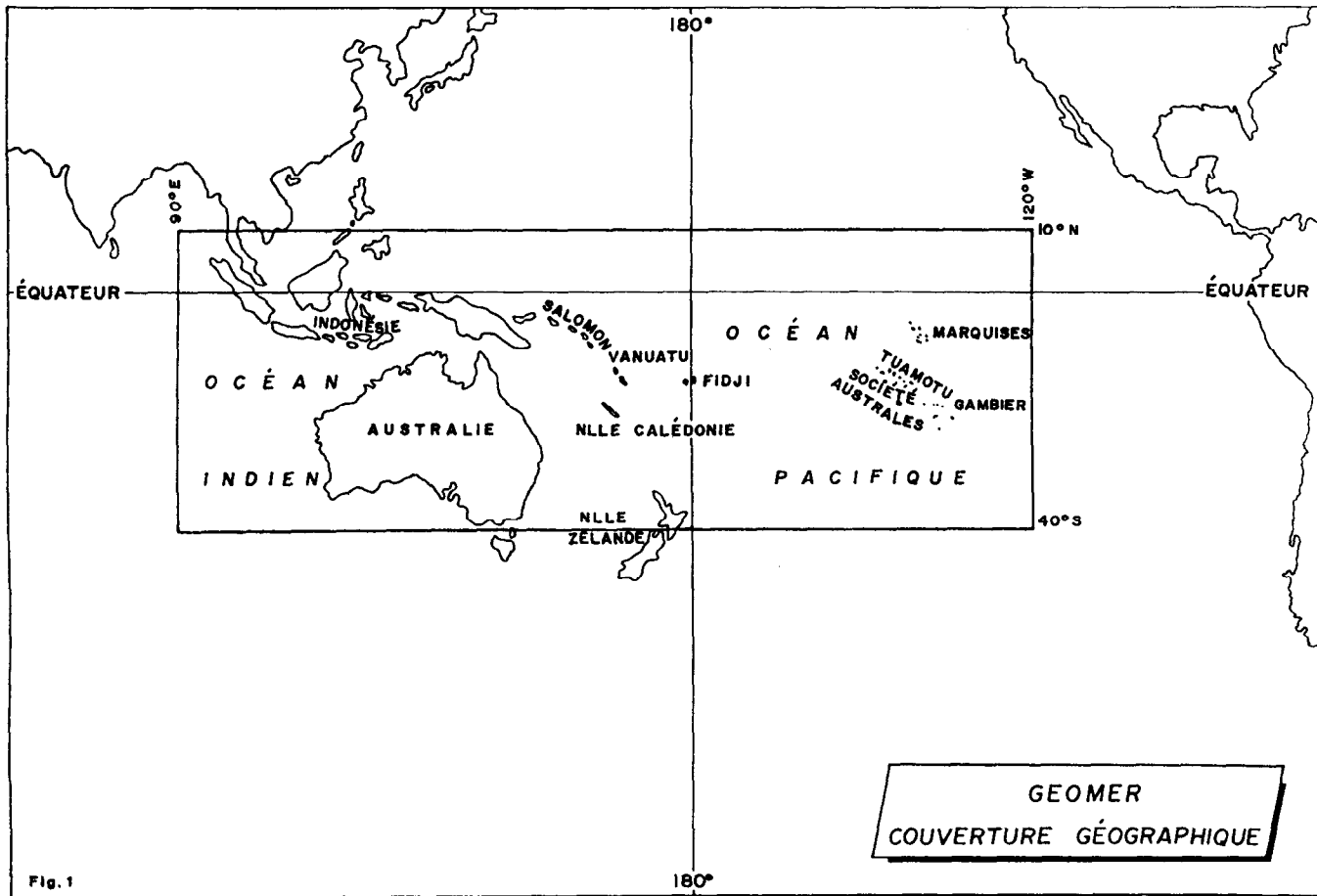


Fig. 1

GEOMER INCORPORATION SEQUENCE 1

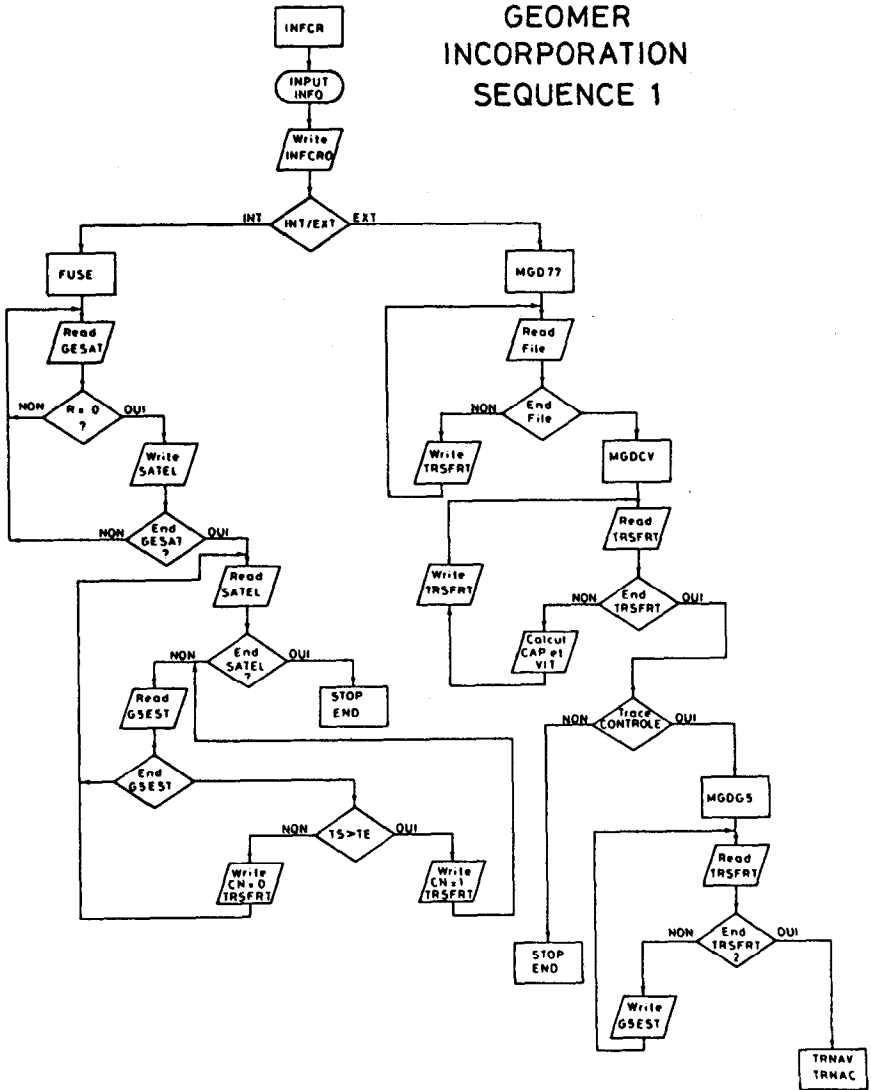


Fig. 2

- de savoir grâce à l'état d'un pointeur s'il y a eu d'autres traversées de cette station soit par la même croisière, soit par une autre croisière.

Ce dernier pointeur permet d'accéder directement à l'adresse où est répertoriée la traversée suivante.

PROFIL contient la liste de tous les profils répertoriés. Chaque enregistrement correspond à un profil et contient les informations permettant d'identifier la croisière et le profil et de situer ce dernier dans le temps et l'espace grille.

STATN contient le descriptif de toutes les traversées de station répertoriées. Un enregistrement correspond à une traversée de station. Il contient les informations permettant d'identifier la croisière et la traversée ainsi que la situation dans le temps et l'espace grille de cette dernière.

CROISR contient tous les renseignements scientifiques, techniques et administratifs de chaque croisière incorporée. Ceux-ci indiquent les noms de la croisière, du navire utilisé, du chef de mission et de la zone géographique couverte, le pointeur de croisière, la situation dans le temps et dans l'espace grille de la croisière, la liste descriptive des moyens mis en oeuvre et la liste des bases gravimétriques sur lesquelles la croisière est calée.

3.2. Organisation et principe de fonctionnement

Incorporation L'incorporation d'une croisière se réalise selon les phases suivantes :

- allocations des pointeurs de croisière, de station, de profil et de traversée de station ;
- création du fichier MESxxx ;
- activation des réseaux de pointeurs ;
- identification du premier point de mesure ;
- positionnement du point sur l'espace grille ;
- recherche de la station support ;
- identification de la traversée de station ;
- positionnement et notation de la traversée ;
- allocation des pointeurs de croisière et de station au point traité ;
- mise à jour des pointeurs dans INFOG ;
- mise à jour des fichiers STATN, PROFIL, DGNDX et DGLST ;
- incorporation du point au fichier MESxxx ;
- traitement du point de mesure suivant.

Quand tous les points de mesure sont traités et incorporés :

- mise à jour du fichier CROISR.

Extraction Lorsqu'une opération d'extraction est lancée, le mode d'extraction est laissé au choix de l'utilisateur. Quel que soit le mode choisi, fenêtre géographique, croisière ou profil, l'opération finale est la même et seuls la sélectivité et le temps de recherche varient.

Une extraction se déroule selon les phases suivantes :

- détermination des stations à consulter ;
- identification de toutes les traversées de ces stations ;
- positionnement des traversées dans l'espace grille ;
- élaboration d'une liste des traversées sélectionnées, tri et mise en ordre chronologique de croisière et de traversée ;
- recherche de tous les pointeurs permettant un accès direct aux données à extraire des différents fichiers MESxxx ;
- positionnement des points de mesure dans l'espace grille ;
- extraction des données ;
- transfert sur bande magnétique.

3.3. Positionnement des données dans l'espace-grille

De toutes les opérations et phases d'opération citées, seules celles concernant le positionnement dans l'espace grille, soit des traversées de station, soit des points de mesure, sont d'un intérêt particulier et mettent en valeur l'ossature de la base sous la forme d'une grille géographique et les relations interactives de deux réseaux de pointeurs.

Incorporation (Figure 3) La position d'un point de mesure dans l'espace géographique permet de situer celui-ci sur la grille GEOMER et sur une station déterminée.

Quand la station support est déterminée, on connaît le numéro de l'enregistrement de DGNDX où elle est répertoriée, autrement dit, on la situe sur l'espace grille et les séquences suivantes sont exécutées :

- lecture de DGNDX ;
- si le pointeur $INDX = 0$, la station est vierge et n'a jamais été traversée par une croisière ;
- la nouvelle traversée est identifiée à l'aide du pointeur de croisière et de numéro chronologique de traversées de station au cours du déroulement de la croisière. Ce dernier est automatiquement incrémenté à chaque changement de station ;
- la traversée est située dans l'espace grille en attribuant à $INDX$ la valeur de $PLST$ le pointeur de liste des traversées ;
- la traversée est répertoriée sur $DGLST$ à l'adresse $PLST$ après assignation du pointeur de station suivante $PSTS$ à la valeur zéro, la traversée étant pour l'instant unique ;

GEOMER INCORPORATION SEQUENCE 2

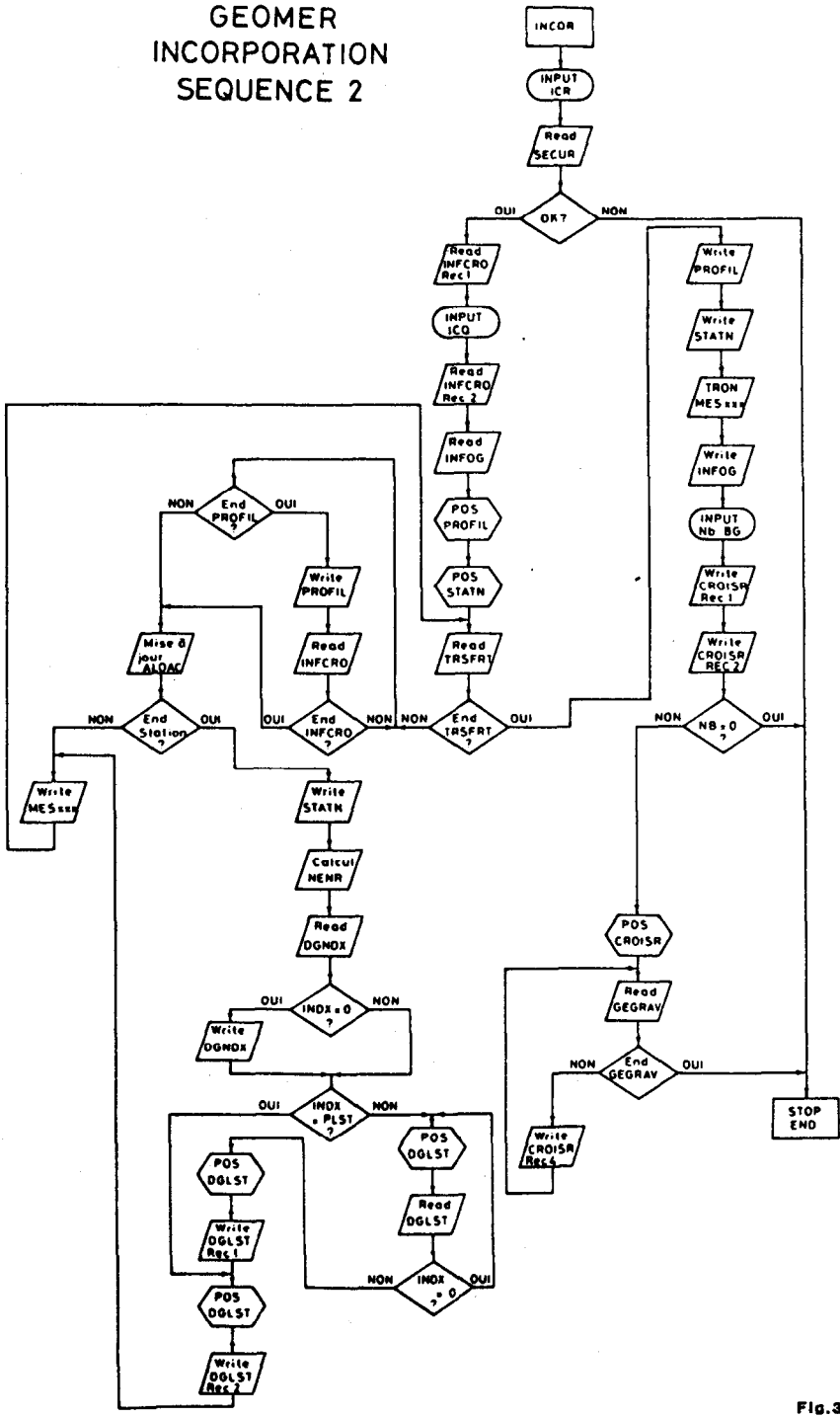


Fig. 3

- si $INDX$ est différent de 0, la station a déjà été traversée par une croisière et la valeur de $INDX$ indique l'adresse de $DGLST$ à laquelle est répertoriée la première traversée. Les séquences suivantes sont alors réalisées : identification de la nouvelle traversée et lecture de $DGLST$ à l'adresse $INDX$.

Si $PSTS=0$ il n'y a eu qu'une seule traversée : la valeur de $PLST$ est assignée à $PSTS$, la traversée est répertoriée à l'adresse $PLST$ avec mise à zéro du nouveau $PSTS$, $PLST$ est incrémenté et il y a passage au point suivant.

Si $PSTS$ est différent de zéro, il y a une autre traversée de répertoriée à l'adresse $PSTS$: $DGLST$ est lu à l'adresse $PSTS$. Si le nouveau $PSTS$ lu est différent de zéro, la lecture continue comme précédemment jusqu'à ce que l'enregistrement portant $PSTS=0$ soit trouvé, autrement dit, jusqu'à ce que toutes les traversées répertoriées aient été identifiées. Lorsque l'on a trouvé la dernière traversée de la station répertoriée, on attribue à $PSTS$ la valeur $PLST$, la traversée est répertoriée à l'adresse $PLST$ avec $PSTS=0$, $PLST$ est incrémenté et il y a passage au point suivant.

Quand le point de mesure suivant a été acquis, son appartenance à la même station est contrôlée. Si ce contrôle est positif le point se voit attribuer les mêmes identificateurs que le point précédent et il ne sera alors particularisé que par son temps au sein d'une même traversée. Si le contrôle est négatif, toutes les opérations précédentes sont à nouveau exécutées.

Extraction (Figure 4) La phase la plus importante de l'extraction est la recherche de toutes les traversées de station réalisées au cours d'une croisière ou d'un profil ou dans une fenêtre géographique.

Dans le cas d'une croisière ou d'un profil, la liste des traversées est réalisée par simple consultation des fichiers $CROISR$, $PROFIL$ et $STATN$.

Dans le cas d'une fenêtre géographique, il s'agit de rechercher toutes les traversées de station, quelles que soient les croisières au cours desquelles elles ont été exécutées. Autrement dit, il faut accéder à toutes les données existantes entrant dans le cadre de la fenêtre, sans aucune omission possible. Pour cela, un balayage de la fenêtre station par station est nécessaire. Pour chaque station entrant dans le cadre de la fenêtre les opérations suivantes sont exécutées : identification de la station et lecture de $DGNDX$ à l'adresse $NREC$.

Si $INDX=0$, station vierge et passage à la station suivante. Si $INDX$ est différent de 0, il y a au moins une traversée : lecture de $DGLST$ à l'adresse $INDX$ et extraction de la traversée. Si $PSTS=0$, il n'y a pas d'autre traversée (passage à la station suivante). Si

GEOMER
EXTRACTION
SEQUENCE 1

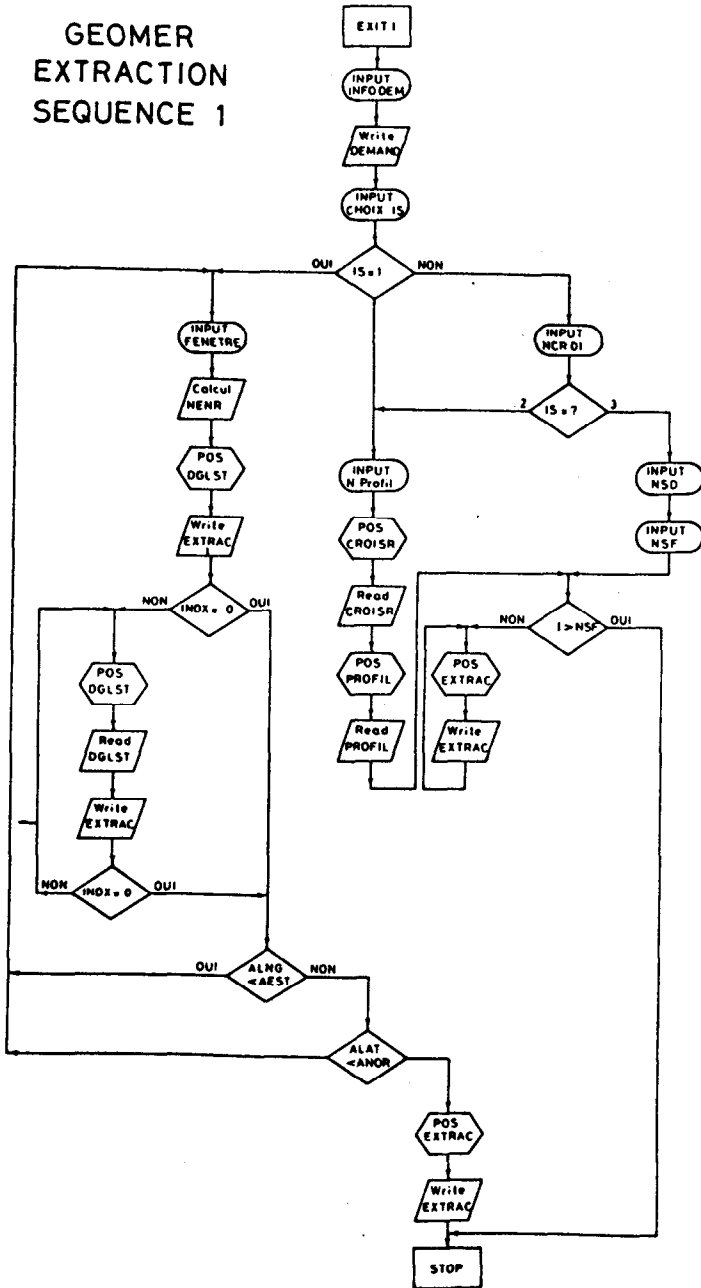


Fig. 4

GEOMER EXTRACTION SEQUENCES 2 & 3

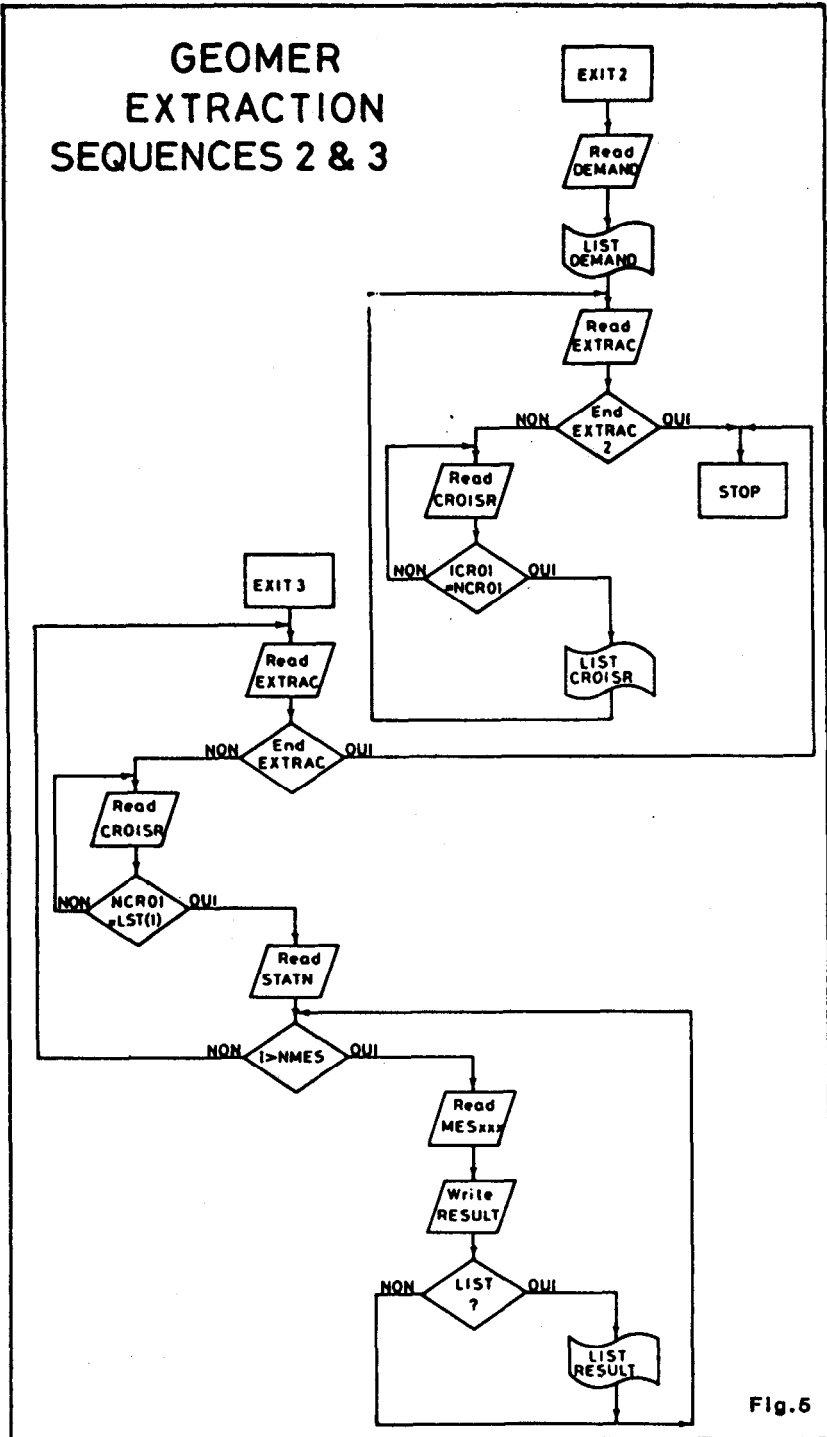


Fig.5

PSTS est différent de 0 il y a encore, au moins une traversée : accès à la traversée suivante à l'adresse PSTS et extraction de cette traversée ; répétition de cette séquence jusqu'à ce que toutes les traversées répertoriées aient été identifiées et extraites ; finalement passage à la station suivante.

Il n'est pas possible d'arrêter la répétition de cette dernière séquence tant que toutes les traversées n'ont pas été identifiées et extraites.

Lorsque la liste de toutes les traversées est établie, celle-ci est triée et les traversées sont classées par ordre chronologique des identificateurs de traversée.

Le premier terme de l'identificateur de traversée permet d'identifier le fichier MES_{xxx} dans lequel sont stockés les points de mesure à extraire.

Le second terme de l'identificateur donne l'adresse du fichier STATN où pourra être lu, d'une part, le pointeur d'accès au premier point de mesure à extraire et d'autre part le nombre de points de mesure répertoriés au cours de la traversée.

Quand tous les points de mesure d'une traversée de station sont extraits et répertoriés dans le fichier RESULT, on passe à la traversée suivante jusqu'à épuisement de la liste.

4. UTILISATION

Comme pour toutes les bases de données, l'accès en écriture est protégé et l'administrateur de la base ou des opérateurs agréés sont les seuls à pouvoir réaliser les opérations d'incorporation.

Les opérations d'extraction, telles qu'elles viennent d'être décrites sont totalement transparentes pour l'utilisateur. Celui-ci n'a qu'à suivre une procédure simple et limitée à la saisie des informations définissant l'extraction à réaliser.

Le transfert des données extraites, sur bande magnétique, est ensuite réalisé soit en format interne soit en format MGD77 si le demandeur est extérieur.

Les demandes d'extraction provenant de l'extérieur sont réalisées par un opérateur agréé.

Tous les échanges de données se font sous le format international d'échange de données de géophysique marine MGD77 mis au point par le *National Geophysical and Solar-terrestrial Data Center Boulder Colorado* dépendant de la *National Oceanic and Atmospheric Administration de l'U.S. Dpt of Commerce*.

Deux phases de développement sont actuellement à l'étude.

Phase 1 Dans un premier temps les logiciels gestionnaires de GEOMER vont être traduits en FTN77 MicroSoft version 3.31 afin de leur assurer une portabilité aussi grande que possible.

Un logiciel de création de base de données à la carte est en cours d'étude. Celui-ci va permettre à un éventuel acquéreur de ce type de base de données spatio-temporelle, de structurer sa propre base en fonction de la couverture géographique souhaitée, de ses besoins et du type de données à gérer.

L'implantation d'une telle base de données pourrait alors se faire sur des micro-systèmes du type compatible IBM PC-AT disposant d'une mémoire de masse de 30-40 MO, de 512 KO de mémoire vive et d'un système de sauvegarde du type steamer.

Phase 2 L'implantation sur un mini-système du type station Sun ou Apollo permettrait d'une part des accès en E/S par réseau type Ethernet et d'autre part des échanges de données avec les grandes bases internationales par connexion à un réseau international de télécommunications.

Certains types de données comme celles de géophysique marine, nécessiteraient alors la mise au point d'un système de protection équipé d'une mise à jour périodique, interdisant l'accès aux données frappées de confidentialité à un interrogateur extérieur non autorisé.

LA GESTION DES DONNEES DES LABORATOIRES D'ANALYSES PHYSICO-CHIMIQUES. EXEMPLE D'APPLICATION : MIDAS

Marc Pansu
(Chimie)
(Laboratoire "Matières organiques")

Centre ORSTOM de Bondy
70/74 route d'Aulnay
93140 Bondy (France)

RESUME - L'information correspondant aux analyses physico-chimiques est très diverse (Figure 1). Elle commence à la définition d'une expérience ou la collecte de renseignements qualitatifs correspondant à un dossier analytique, se poursuit par le choix de mesures associées à des protocoles analytiques avec éventuellement des mises au point méthodologiques. La partie laboratoire comprend la préparation d'échantillons préalablement à des mesures où des données brutes sont collectées à la sortie des appareillages. Il faut alors réduire ces données en résultats définitifs qualitatifs ou quantitatifs et contrôler ces résultats.

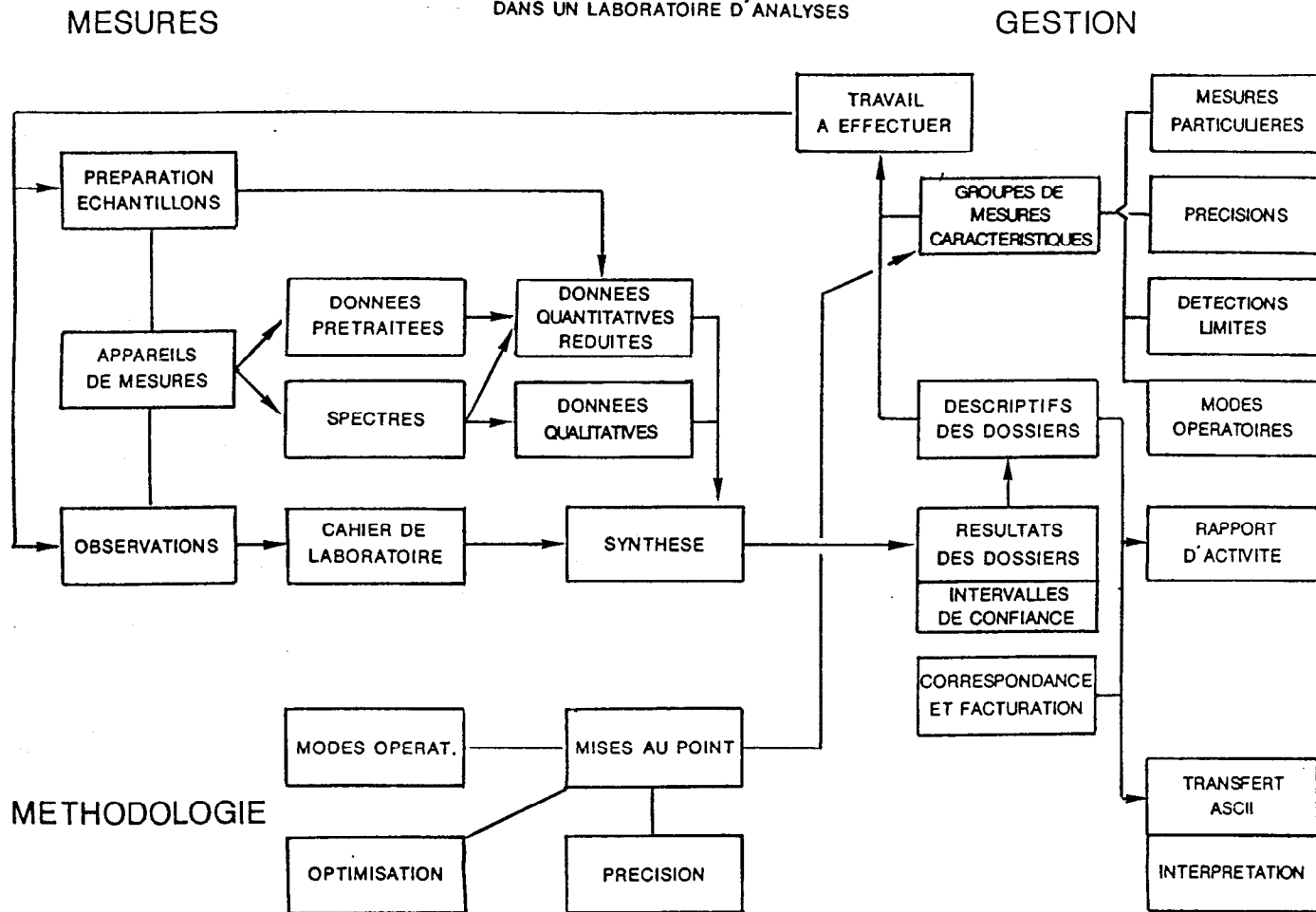
Le logiciel MIDAS (Mixage de l'Information concernant les Données Analytiques et Spectrographiques) tente de répondre à l'ensemble des problèmes de saisie et de gestion de cette information physico-chimique, tant dans la définition du travail que dans sa mise au point, sa réalisation et son contrôle.

INTRODUCTION

La diversité de l'information analytique engendre une panoplie des besoins informatiques d'un laboratoire d'analyses assez étendue : gestion du laboratoire, traitement de signaux, gestion des résultats analytiques, traitement des données, chimométrie, textes, etc. Le chimiste moderne devient donc un grand "*consommateur*" de logiciels tels que traitements de textes, gestionnaires de bases de données, tableurs, calculs statistiques, programmes spécialisés de chimométrie.

Il n'existe pourtant pas dans les produits de grande diffusion, de logiciel prenant vraiment en compte tant la spécificité de la mesure physico-chimique que celle du travail de laboratoire. Cette constatation nous a conduit à la définition du logiciel MIDAS, base

Figure 1 : LA CIRCULATION DES DONNEES
DANS UN LABORATOIRE D'ANALYSES



de données pour les informations physico-chimiques (Figure 1) sur laquelle peuvent se greffer en amont des applications concernant les plans d'expériences, le traitement du signal, les calculs d'étalonnages et en aval, des programmes statistiques et d'aide à l'interprétation.

1. DESCRIPTION GENERALE DU LOGICIEL MIDAS (Fichier d'aide 1)

MIDAS = Mixage de l'Information concernant les Données Analytiques et Spectrographiques :

- les échantillons sont regroupés en dossiers correspondant à un même type d'étude. Chaque dossier peut contenir un seul ou un nombre non limité d'échantillons. Les descripteurs du dossier ainsi qu'un texte associé peuvent être saisis dès la décision d'un projet d'études, lorsqu'on ne connaît encore ni les échantillons qui seront analysés, ni les dosages qui seront choisis. Chaque dossier est rangé dans une (ou plusieurs) bibliothèque(s) propre au laboratoire. Chaque bibliothèque MIDAS est automatiquement répertoriée dans une "directory" ;

- Les autres informations concernant le dossier sont alors saisies au fur et à mesure de leur connaissance : Descriptifs des échantillons, variables à mesurer, état des mesures, préparation des échantillons, saisie des résultats et observations dans un cahier de laboratoire ;

- Les descripteurs des variables sont optimisés à des variables physico-chimiques : identifiant, descriptions analytiques, unité, limite de détection, précision, valeurs permettant le calcul des intervalles de variation, prix.

Une originalité de MIDAS consiste en sa personnalisation pour chaque laboratoire : le programme MIDINST permet de ranger les variables déterminables par ce laboratoire par groupes (de une à un nombre non limité de variables) dans une (ou plusieurs) bibliothèque(s) d'installation. On pourra ainsi installer des groupes comme : éléments majeurs des roches, oligo-éléments des plantes, bases échangeables sols, soufre dans sédiments, humus bondy classique, polysaccharides des sols, etc. La saisie des analyses à effectuer sur un dossier est alors très rapide puisqu'elle consiste en un simple transfert de tous les descripteurs des variables choisies dans un ou plusieurs groupes d'installation. Des textes "modes opératoires" peuvent être associés à chaque groupe et consultés à tout moment.

Avec chaque analyse sont automatiquement associés deux fichiers de nombres réels : un fichier mesures et un fichier laboratoire. Chaque fichier mesures se dimensionne automatiquement à

la taille du fichier échantillon. Chaque mesure peut avoir trois états : "demandé", "non demandé" ou résultat réel. Chaque fichier laboratoire de taille non limitée permettra la saisie et l'archivage de données en sorties des appareils (spectres, chromatogrammes, hauteurs de pics, etc.).

Les procédures de lecture du dossier permettent à chaque opérateur d'avoir un état complet du travail à effectuer.

Peut alors commencer le travail lui-même de laboratoire. Le logiciel permet ici encore un découpage du travail tel qu'il est effectué en séparant la préparation des échantillons et les analyses :

- la préparation des échantillons permet la saisie des pesées (interface RS232 unidirectionnelle METTLER) et des dilutions ou concentrations associées à chaque échantillon, ces chiffres étant alors transformables en facteurs multiplicatifs ;

- la partie travail de laboratoire permet alors à l'opérateur de saisir soit des mesures définitives, soit d'apporter une correction interactive par les facteurs à ses mesures lors de leur saisie au clavier. Elle doit permettre avec des développements spécifique à chaque appareil (interfacage, traitement du signal, calculs d'étalonnages, etc) de réduire les données collectées dans chaque fichier laboratoire en données définitives dans les fichiers mesures correspondants (ou dans le texte cahier de laboratoire s'il s'agit de résultats qualitatifs).

Une procédure de conversion des fichiers mesures en ASCII permet l'utilisation en aval de logiciels statistiques tels que STATGRAPHICS, de tableurs et traitements de textes.

Une procédure d'exploration d'une bibliothèque permettra au responsable d'avoir à la demande l'état d'avancement d'un dossier, de l'analyse d'un échantillon, d'une analyse particulière dans les dossiers d'un client, etc.

Le logiciel correspond à une redéfinition du logiciel SPCLAS (M. Pansu, ORSTOM, 1983) écrite en Basic HP. Il est écrit en Turbo PASCAL sur système MS-DOS compatible IBM PC/AT. Son utilisation est possible en réseau (rejet si présence d'un autre utilisateur) ou individuellement avec disque dur ou double lecteur de disquettes.

Il comporte six types de fichiers pour le stockage des données : fichier directory, fichier bibliothèque, fichier échantillons, fichier variables, fichiers mesures, fichiers laboratoires, fichiers textes. La déclaration de types au début de programmes Turbo-PASCAL permet aisément de changer la longueur d'un champ ou d'ajouter un champ à un fichier pour gérer des informations supplémentaires.

Les fichiers textes servent à trois usages :

- aide ponctuelle à l'utilisation du logiciel et à des développements (noms et structure des fichiers de données fournis) ;
- aide à l'analyse chimique par consultation de modes opératoires analytiques associés aux groupes de la bibliothèque d'installation ;
- textes associés à chaque dossier : cahier de laboratoire et correspondance.

Le logiciel facilite la réalisation des principaux calculs de laboratoire (choix des calculs par fenêtres d'écran) associés aux contrôles des données et mises au point méthodologiques.

Les programmes comportent un fichier de commande .com inséré dans un fichier .bat ainsi que cinq fichiers compilés .chn. A l'exécution, le chargement d'un fichier .chn est instantané et les données communes restent en mémoire d'un programme à l'autre.

Les choix de travail, de transferts de variables, etc sont réalisés par menus en version couleur si disponible. Les seules touches actives du clavier sont alors les flèches pour la sélection et les touches de fonction indiquées. Les données devant être entrées au clavier sont associées à des gestions de curseur plein écran et des procédures de correction/effacement.

2. GESTION DES INTITULES DE BIBLIOTHEQUE (fichier d'aide 2)

Cette partie permet de ranger les dossiers analytiques dans une (ou plusieurs) bibliothèques correspondant au laboratoire.

Il faut donc au préalable créer sa bibliothèque de travail (1) en lui donnant un nom de 8 caractères maximum.

Cette bibliothèque est alors immédiatement répertoriée dans un fichier nommé DIRECTOR.BIB de structure :

```
fichierdirectory = record
                    classement      : integer ;
                    ficbibliotheque : ch8 ;
```

end ;

A chaque nom de bibliothèque (ficbibliotheque) est affecté un numéro entier de classement.

Lors du fonctionnement ultérieur du logiciel, le numéro de classement sera automatiquement transformé en caractère alphanumérique avec la procédure :

```
existantbib(ficbibliotheque:ch8, numerobib:char, exist:boolean)
du fichier : existbib.pas.
```

Le nombre de bibliothèques est limité à 36. Au-delà, il faut changer de répertoire. Pour une utilisation rationnelle du logiciel, il n'est conseillé de créer une nouvelle bibliothèque, que lorsqu'on aura rempli de dossiers la précédente.

Pour effacer une bibliothèque, utiliser la procédure 'Opérations sur fichiers' du menu principal.

A chaque bibliothèque, correspond un fichier : `ficbibliotheque.lab`, de structure :

```
fichierbiblio = record
    nomdossier      : ch8 ;
    numdossier      : char ;
    datarrivee      : ch8 ;
    datefin         : ch8 ;
    originegeo      : ch50 ;
    interlocut      : ch30 ;
    methodes        : ch50 ;
    operateurs      : ch20 ;
    substrat        : ch200 ;
    fichvar         : ch12 ;
    fichech         : ch12 ;
```

end ;

Dans ce fichier, on va ranger les renseignements correspondant à chaque dossier, lors d'un projet d'études sans qu'il soit alors nécessaire de connaître les références des échantillons ni les analyses exactes à effectuer.

On peut en outre avec l'éditeur Turbo ou un traitement de textes type Wordstar, (Option N ou Wconvt B : conversion des fichiers en ASCII) saisir un texte qui sera toujours associé au dossier, dans un fichier :

```
nomdossier.bi numerobib
```

Par défaut, la date de début (`datarrivee`) sera automatiquement celle de l'ordinateur (mettre à jour la date du MS-DOS).

Pour effacer un dossier : on peut utiliser la dernière option de ce premier menu si aucune référence d'échantillons n'a été saisie. Sinon, utiliser l'option 'Opérations sur fichiers de données' du menu principal. Avec la même option, il est possible de copier un dossier dans une autre bibliothèque.

A chaque dossier est affecté automatiquement un caractère alphanumérique (`numdossier`) qui servira, avec `numerobib`, à désigner les fichiers ultérieurs qui lui correspondent (échantillons, variables, mesures, laboratoire) en évitant les confusions.

Une bibliothèque peut contenir au maximum 50 dossiers.

3. UN DOSSIER DE LABORATOIRE EST CONSTITUE PAR (fichier aide 3)

1) *Un plan d'expériences (1)*: optimisation de méthodes, étude de causes d'erreurs, etc. ;

2) *Un ensemble (ou un seul) d'échantillons* sur lesquels devront être réalisés différentes mesures.

Dans ce cas, saisissez dans l'ordre :

- les références et caractéristiques des échantillons (2) ;
- les mesures à effectuer sur chaque échantillon (3).

L'option (4) permet une première visualisation du travail à effectuer sur le dossier.

L'ensemble des informations (travail à effectuer, caractéristiques des échantillons, des variables, texte, modes opératoires, etc) est accessible avec l'option "Lecture sur dossiers" du menu principal. Elles constituent les premiers renseignements du cahier de laboratoire et permettent à l'utilisateur de commencer le travail analytique.

Au cours du travail, il modifiera les informations numériques et alphanumériques correspondant au dossier : préparation échantillons, saisie résultats, observations sur les échantillons, sur les variables, observations générales, etc..

Les observations générales seront notées sur le fichier :
nomdos.bi numerobib

Les observations concernant les échantillons et variables seront corrigées ou ajoutées avec le présent menu. Les informations correspondant à la préparation des échantillons et mesures seront saisies avec l'option "Travail de laboratoire" du menu principal.

Le logiciel peut donc remplacer complètement le cahier de laboratoire ainsi que le dossier correspondant aux protocoles analytiques.

3) *Gestionnaire des échantillons (fichier d'aide 3-2)*

Un fichier échantillon est relié à un dossier d'une bibliothèque par son nom soit :

nomdossier .E numerobib numerodos

Sa structure est la suivante :

```
fichierchantillon = record
    echantillon : ch10 ;
    facteur1    : real ;
    facteur2    : real ;
    descript    : ch80 ;
    cord1       : real ;
    cord2       : real ;
    cord3       : real ;
end ;
```

Les touches de fonction indiquées vous permettent :

<F2> *saisie* : ajoute en fin de fichier une référence de 10 car.
maxi à echantillon

met facteur1 et facteur2 à 1
met un blanc dans descript
initialise cord1 cord2 et cord3 à -32767

<F3> *correction* : permet de modifier la référence echantillon
à la position du curseur

<F4> *effacement* : efface l'enregistrement du curseur de
fichierchantillon et efface toutes les mesures correspondantes s'il y
en a

<F5> *insertion* : non installée

<F6> *description* : saisie ou correction du texte descript de 80
caractères associé à l'échantillon du curseur.

Curseur : seules les flèches haut et bas sont actives ainsi que
les deux flèches habituelles de changement de page.

4) *Gestion des variables. (fichier d'aide 3-3)*

Un fichier variable est associé à un dossier d'une bibliothèque
par son nom :

nomdossier .V numerobib numerodos

Sa structure est la suivante :

```
fichiervariables = record
    composant           : ch6 ;
    mesures             : ch12 ;
    descripteurunite    : ch6 ;
    descripteurprecision : integer ;
    limitedetection     : real ;
    coefvariation       : real ;
    nbrepetition       : integer ;
    nbressaimoyen      : integer ;
    descript1           : ch80 ;
    descript2           : ch200 ;
    descript3           : integer ;
    descript4           : real ;
    descript5           : real ;
    exp                 : array[1..6] of real ;
end ;
```

- descripteurprecision = nombre de décimales pour l'affichage des résultats ;
 - coefvariation = 100 écart-type/moyenne ;
 - nbrepetition = degrés de libertés de la mesure du coefficient de variation ;
 - nbressaimoyen = nombre répétitions mesures d'une variable ;
 - descript1 = nom de la variable ;
 - descript2 = principe de la mesure ;
 - descript4 = prix de l'analyse.

Il ne peut y avoir dans un même dossier deux variables portant le même nom (composant).

Les variables peuvent être saisies :

- directement (2^o option) : touche <ENTER> pour le positionnement. Pour qu'une saisie soit effective, il suffit d'avoir donné une valeur au nom de variable (composant). Cette option sert également à la correction en introduisant le nom de variable à corriger comme premier descripteur. Pour corriger ce nom, effacer la variable (3^o option) et recommencer. Les valeurs par défaut des champs numériques sont -32767, celles des alphanumériques, un blanc ;

- depuis une bibliothèque d'installation : cette procédure est de loin la plus rapide et la plus fiable. Il faut avoir au préalable utilisé MIDINST pour saisir tous les descripteurs des analyses du laboratoire rangées dans des groupes. Le même programme permet à tout moment de modifier ces descripteurs analytiques. L'option 'Variables installées' permet de transférer dans le dossier la variable choisie par le curseur avec tous ses descripteurs.

A chaque variable est associé un fichier mesures :
 composant .M numerobib numerodos
 soit un fichier de nombres réels dimensionné à la taille du
 fichier échantillon.

Chaque mesure peut avoir trois états :

D = demandée = -32767

N = non demandée = -32766.

nombre réel = chiffre de mesure.

Par défaut, à la saisie des variables, toutes les mesures sont à
 D. L'option 'Etat des mesures' permet de sélectionner avec le cur-
 seur les mesures de même état et d'indiquer cet état avec N ou D
 pour chaque échantillon sur un éditeur plein écran.

A chaque variable est également associé un fichier à utiliser
 pour le travail de laboratoire :

composant .L numerobib numerodos

soit un fichier de réels de longueur non limitée.

4. TRAVAIL DE LABORATOIRE (Fichier d'aide 4)

Permet de distinguer la préparation des échantillons : pesées
 et dilutions (Option 1), de la saisie des mesures :

- corrigées : sur éditeur plein écran pour chaque
 variable (2) ;

- avec correction par les facteurs résultant de la prépa-
 ration des échantillons (3)

- les deux autres options sont à développer spécifique-
 ment avec chaque appareil de mesures.

Les mesures (spectres, chromatogrammes, résultats calculés,
 etc.) seront saisies de manière séquentielle dans chacun des fichiers
 correspondant à chaque variable, nommés :

Nomvar .L numerobib numerodos

Des programmes devront alors être employés pour réduire les
 données dans les fichiers :

Nomvar .M numerobib numerodos

contenant les résultats définitifs quantitatifs et/ou dans le
 fichier nomvar.BI numerobib pour les observations qualitatives.

Préparation des échantillons. (fichier d'aide 4-1)

- saisies pesées et dilutions d'échantillons. Opérations
 arithmétiques simples sur ces facteurs pour les transformer en
 facteurs de correction des mesures. Par défaut les facteurs sont
 unitaires ;

- saisie manuelle : Editeur analogue à saisies manuelles
 mesures ;

- interfacages : Avec Balance METTLER AE sortie de donnée RS232 ref 011. Même éditeur que précédemment, avec déplacement à l'échantillon choisi par <ENTER>. Presser la barre <ESPACE> ou une autre touche pour mise en attente de pesée. La balance n'envoie alors la pesée que lorsqu'elle est stable. Après cet envoi, le curseur passe à l'échantillon suivant.

Si l'ordinateur n'est pas connecté, la touche <ESPACE> ne bloque le logiciel que pendant 10 secondes (sur Goupil G40).

- Interfaces avec autres balances ou diluteur : Modifier la procédure Balance.pas selon le format de sortie des données.

5. LECTURE INFORMATION D'UN DOSSIER (Fichier d'aide 5)

1) Lecture d'un dossier :

En tête = information de la bibliothèque.

Variables = 6 colonnes par écran avec indication du nombre total d'écrans et du numéro d'écran actuel.

Echantillons = 17 lignes par page avec gestion des pages analogue à celle des écrans.

<ENTER> = Ecrans suivants puis pages suivantes.

Affichage des données :

- D = mesure en attente.

- Blanc = mesure non demandée.

- Valeurs numériques : avec nombre de décimales correspondant à descripteurprecision de fichiervariables.

Avec indication LD si valeur inférieure à limitedetection de fichiervariables.

2) Descriptifs échantillons : Lecture par page des descripteurs (descript de fichierechantillon)

3) Signification des variables : Lecture par page des noms (descript1 de fichiervariables)

4) Descriptifs : Lecture par pages des principes des méthodes de mesure des variables (descript2 de fichiervariables)

5) Relation bibliothèque d'installation et modes opératoires : trouve le groupe d'installation éventuel de chaque variable du dossier. Permet de lire :

- les descriptifs des groupes d'installation : nom, date mise au point, principe, etc. ;

- les modes opératoires associés à ces groupes accompagnés du nom du fichier pour leur saisie/correction.

6) *Intervalles de confiance des mesures d'un dossier* : indication de l'intervalle entourant les valeurs numériques listées en 1) selon le niveau de probabilité $\alpha/2$ choisi et rappelé dans l'en-tête. Avec les informations de fichiervariables (si elles ont été saisies), le logiciel calcule :

- valeur de t de Student pour la probabilité indiquée et (nbrerepetition -1) degrés de liberté ;

- pour chaque valeur mesurée :

$$s = \text{coeffvariation} \times \text{mesure} / 100$$

$$\text{Int} = s \times t / \sqrt{\text{nbressaimoyen}}$$

7) *Facturation d'un dossier* : pour chaque variable, si descript4 de fichiervariables a été saisi :

- affichage par page de la variable et des prix correspondant au nombre total de mesures demandées et au nombre de mesures effectuées ;

- affichage en fin de page du coût total du dossier et du coût des analyses effectuées.

8) *Texte associé au dossier* : lecture par page du texte en mode fichier ASCII correspondant aux remarques du cahier de laboratoire et aux correspondances. Même en l'absence de texte, le logiciel donne Le nom du fichier à utiliser pour sa saisie.

6. CALCULS DE LABORATOIRE (fichier d'aide 6)

Une calculatrice vectorielle sur les variables chimiques

L'opérateur construit sa formule de calcul au moyen de deux fenêtres d'écran : celle de gauche visualise la liste des symboles des variables du dossier, celle de droite la liste des calculs disponibles. Les flèches permettent le déplacement alternativement dans l'une ou l'autre des fenêtres. Elles autorisent les choix sur un nombre non limité de variables par défilement continu dans une fenêtre de quinze symboles.

Le choix de la variable ou du calcul à la position du curseur en inverse vidéo est réalisé par pression de la touche de fonction <F6>, la sortie par la touche <F10>.

Le menu principal permet le choix des options:

- transformation de variables ;
- opération entre variables ;
- calculs sur une variable ;
- calculs entre variables.

Transformation de variables :

Autorise les calculs suivants sur toutes les mesures saisies de variables:

- ajout : ajoute une constante ;
- soustraction : soustrait une constante ;
- multiplie : multiplie par une constante ;
- divise : divise par une constante non nulle ;
- log : calcule les logarithmes népériens pour les valeurs positives ;
- exp : calcule les exponentielles ;
- carre : élève au carré ;
- racine : donne la racine carrée des valeurs positives ou nulles ;
- inverse : donne l'inverse des valeurs non nulles ;
- puissance : élève à la puissance indiquée.

Lorsqu'une mesure est à l'état "demandé" ou "non demandé" (blanc), le résultat du calcul est "non demandé" (blanc).

La formule de calcul s'inscrit en bas de l'écran à mesure des choix. Par exemple:

choix +	Formule
variable TEMP	TEMP
calcul : ajout	TEMP+
nbre à ajouter ?273	TEMP+273
calcul : multiplie	(TEMP+273)*
multiplicateur ?40	(TEMP+273)*240
calcul : inverse	1/((TEMP+273)*240)
calcul : exp	EXP(1/((TEMP+273)*240))

A chaque calcul, sont mis à jour les paramètres chimiques de la variable choisie : soit y le vecteur transformé par f du vecteur x ($y = f(x)$). Les lois de propagation des erreurs donnent :

$$\sigma_y = \left| \sigma_x \frac{dy}{dx} \right|, \text{ avec}$$

y = écart type de la moyenne y
x = écart type de la moyenne x

Le coefficient de variation CVx devient

$$CV_y = \text{ABS}(x \text{ CV}_x \frac{dy}{dx}).$$

Des calculs analogues mettent à jour limite de détection et précision.

Les résultats du calcul correspondant à la formule du bas de l'écran sont affichés après la sortie par <F10> sur une nouvelle fenêtre d'écran à défilement continu en comparaison avec les valeurs initiales de la variable.

Après une nouvelle pression sur <F10>, le programme demande le symbole éventuel de la variable transformée pour le stockage dans le dossier des résultats associés aux nouveaux paramètres chimiques. La formule correspondante devient le nom de la variable transformée.

En l'absence de symbole (touche <ENTER> ou <F10>), la variable transformée n'est pas mémorisée. En aucun cas la variable source n'est détruite : pour cela on se reporte à la procédure normale d'effacement de variable.

Pour une correction éventuelle des descripteurs, on se reporte également à la procédure de correction d'une variable.

Attention : des changements ultérieurs de la variable source ne mettent pas à jour les calculs. Effacer la variable résultat et recommencer, la procédure est très rapide.

Opérations entre variables :

Ces opérations élémentaires sont les plus courantes au laboratoire. En particulier pour :

- les bilans analytiques ou sommes des résultats de plusieurs variables à comparer avec une constante ou une autre variable ;
- l'étude des rapports de deux variables (carbone/azote, silice/alumine, etc.) ;
- les comparaisons de méthodes par paires : étude de la distribution des différences de deux variables ;
- les moyennes de plusieurs méthodes.

La calculatrice vectorielle fonctionne avec un système à deux fenêtres d'écran analogue à ci-dessus : choix d'une variable, puis de l'opération d'une autre variable, etc.. La formule s'inscrit de même en bas de l'écran.

Les calculs suivants sont disponibles :

- somme : ajoute les résultats d'une nouvelle variable aux résultats courants. Le résultat est blanc si celui d'un des deux opérandes n'est pas une mesure (état "D" ou "ND") ;
- différence : idem, en retranchant les résultats de la nouvelle variable ;
- produit : idem, en multipliant par les résultats de la nouvelle variable ;
- rapport : idem, en divisant par les résultats de la nouvelle variable ;
- somme def : idem à somme, mais le résultat est la mesure existante si l'un des deux chiffres n'est pas une mesure ;
- moyenne def : idem, avec les moyennes des seules mesures effectuées.

La mise à jour des paramètres chimiques est réalisée selon les lois de propagation des erreurs aléatoires:

sommes, différences : $\sigma_y = \sqrt{(\sum \sigma_{xi}^2)}$, où σ_y est l'écart type de la moyenne y et σ_{xi} sont les écarts types des moyennes x_i ;
produits, rapports : $CV_y = \sqrt{(\sum CV_{xi}^2)}$, où CV_y est le coefficient de variation y et CV_{xi} sont les coefficients de variation des x_i .

La procédure d'affichage des résultats sur fenêtre d'écran et de sauvegarde est analogue à précédemment.

Calculs et tests sur une variable :

Procédure de fonctionnement par fenêtre d'écran analogue à précédemment. Ici les calculs ne portent plus sur chacun des résultats des variables mais concernent des paramètres de leur distribution :

effectif : nombre de mesures effectuées sur une variable,
somme : somme des mesures effectuées sur une variable,
moyenne : moyenne des mesures effectuées sur une variable,
médiane : médiane des mesures effectuées sur une variable,
Ectype : écart-type des mesures effectuées sur une variable,
Max : mesure maximum de la population,
Min : mesure minimum de la population,
Outlier : test de DIXON de recherche des valeurs aberrantes,

F/erreur : compare la variance de la distribution à la variance provenant du dosage $s^2 = (\bar{x} * CV/100)^2$. Une valeur de F non significative indique que les variations observées ne sont pas interprétables ;

Xbar=0 ? : test l'hypothèse nulle de nullité de la moyenne : test $t = \bar{x} / n/s$. Très utile pour les comparaisons par paires de séries de résultats (faire la différence des deux variables et appliquer ce test sur la variable résultat) ou la comparaison d'une série de résultats à une valeur étalon (faire la différence entre la série et la constante , puis test sur la variable résultat).

Calculs entre variables :

Ces calculs seront développés en liaison avec la partie "plans d'expériences" de gestion des dossiers : étude de causes d'erreurs et analyses de variances, optimisation numérique, régressions multiples. Ce logiciel ne vise pas à remplacer les logiciels statistiques mais doit faciliter la réalisation des principaux calculs et tests fréquemment nécessaires au laboratoire.

7. EXPLORATION D'UNE BIBLIOTHEQUE (fichier d'aide7)

Cette procédure n'est pas encore installée. Elle le sera en prenant au mieux en compte les besoins des utilisateurs : état d'une analyse dans la bibliothèque, d'un échantillon, des dossiers d'un demandeur, etc.. Son installation est d'ailleurs très facile connaissant la structure des fichiers fournie dans les autres menus d'aide.

8. OPERATION SUR FICHIERS DE DONNEES (Fichier d'aide 8)

1) Répertoire d'un disque : fournit les fichiers en version formatée quatre colonnes selon le critère de sélection.

- *.LAB => toutes les bibliothèques de travail.
- *.INS => toutes les bibliothèques d'installation.
- *.BI* => tous les fichiers textes chimiques
- *.E* => tous les fichiers échantillons.
- *.V* => tous les fichiers variables.
- *.M* => tous les fichiers mesures.
- *.L* => tous les fichiers laboratoires.
- *.COM
- *.CHN
- *.BAT => fichiers programmes
- *.HLP => fichiers d'aide à l'utilisation du logiciel.

2,3) N'utiliser que les procédures d'effacement de ce menu. Des procédures d'effacement concernant la bibliothèque d'installation sont prévues dans MIDINST. Ne pas effacer de fichiers sous MS-DOS. Sécurité : le logiciel empêche d'effacer une bibliothèque contenant des dossiers. Il faut effacer au préalable tous les dossiers (opt.3).

4,5) Ces procédures d'effacement d'une variable et d'un échantillon sont également intégrées dans 'GESTION DES INTITULES DE DOSSIER'.

6) La copie d'un dossier dans une autre bibliothèque est très utile lorsque celui-ci doit être analysé par plusieurs laboratoires ou plusieurs modules d'un laboratoire dans un même lieu. On évite ainsi les répétitions de saisie de l'en-tête et des échantillons.

7) Conversion ASCII des fichiers mesures et des références d'échantillons. Caractères séparateurs LF. Les fichiers sont récupérables par STATGRAPHICS, WORDSTAR, MULTIPLAN, etc.. Cette procédure doit être complétée avec des options pour la conversion plus rapide de toutes les mesures d'un dossier, des identifiants de variables, etc.. Nous devons aussi ajouter l'importation de fichiers ASCII sous MIDAS dans le but d'établir une compatibilité totale de ce logiciel avec des tableurs performants pour la sélection de données et avec les logiciels réseaux de gestion relationnelle de bases de données (10base, dBASE III).

METHODE D'ANALYSE STRUCTUREE POUR LA MISE AU POINT DE LOGICIELS

Patrick Séchet
(Informatique)

(UR 502, Cadres Spatiaux de l'Indépendance Alimentaire)

EMBRAPA

Empresa Brasileira de Pesquisa Agropecuaria
SuperCenter Venâncio 2.000
70333 - Brasília, DF (Brésil)

RESUME - L'évolution actuelle des micro-ordinateurs, de plus en plus puissants et de moins en moins chers, conduit à une diffusion sans cesse croissante de l'informatique, qui envahit rapidement tous les secteurs de l'activité humaine. L'ORSTOM n'échappe nullement à ce phénomène, bien au contraire, de telle sorte que l'on y voit se développer des logiciels dans les domaines scientifiques les plus variés.

Au fur et à mesure que l'informatique étend son champ d'application, elle s'attaque nécessairement à des tâches de plus en plus complexes. Les réalisations artisanales actuelles, fait d'équipes restreintes (traditionnellement des chercheurs isolés), dont l'activité se situe essentiellement au niveau de la phase de programmation, devront progressivement faire la place à des développements de systèmes plus sophistiqués, mieux intégrés, susceptibles de satisfaire à des exigences de fiabilité, disponibilité et pérennité.

Etant entendu que les décisions majeures sont prises dans les étapes initiales, un transfert des efforts de réalisation vers les phases de conception et spécification (encore appelées respectivement analyse fonctionnelle et analyse organique) est indispensable, d'autant plus que la plupart des erreurs qui conduisent à un disfonctionnement du logiciel sont commises lors de ces étapes.

L'adoption d'une méthodologie rigoureuse pour la conception d'applications informatiques, quelles qu'elles soient, constitue un moyen efficace pour améliorer la qualité finale de celles-ci, en agissant essentiellement sur la communication, à plusieurs niveaux :

- au niveau du dialogue entre le concepteur (analyste) qui doit présenter un modèle logique de l'application (qu'il ne connaît généralement pas), et qui lui est décrite par le commanditaire (utilisateur), normalement peu soucieux de cette modélisation conceptuelle ;

- au niveau du dialogue entre l'analyste et le(s) programmeur(s), en fournissant une spécification précise de tous les objets qui seront manipulés dans l'étape suivante de programmation, ce qui se traduira par une diminution du nombre d'erreurs de cette phase ;

- au niveau du dialogue entre analystes, qui peuvent s'appuyer sur un langage commun pour débattre des aspects de représentation, spécifiques à l'application.

La méthode d'analyse structurée de systèmes, introduite par Chris Gane et Trish Sarson en 1977, fournit un ensemble de techniques de représentation graphique pour construire un modèle logique du système. Le schéma est développé par raffinements successifs depuis la vision globale de l'application, jusqu'à la représentation des fonctions élémentaires.

L'outil principal de modélisation est le diagramme de flux de données, qui fournit une vision graphique des relations entre les divers composants de l'application, permettant alors à l'analyste, à l'utilisateur et au programmeur de percevoir les diverses parties du système à partir d'une même structure modulaire.

L'analyse fonctionnelle élaborée dans la première étape de cette méthodologie conduit naturellement à une participation active de l'utilisateur lors de la définition des flux de données, prélude à la construction du dictionnaire de variables. A l'issue de cette phase, un document (le projet logique) est produit et va constituer la base de travail de l'étape suivante de spécification.

L'approbation, par l'utilisateur, de ce document fige la version du système à mettre au point : l'adéquation du produit dépend directement de la bonne compréhension du projet logique par l'utilisateur, d'où l'importance de cette participation.

Il est évident que l'adoption d'une telle méthodologie est un préalable indispensable à l'utilisation de techniques nouvelles relevant du génie logiciel. Celles-ci sont susceptibles d'augmenter substantiellement la productivité de la phase de programmation, goulot d'étranglement traditionnel pour la mise au point des applications informatiques, en offrant un environnement plus approprié et mieux intégré avec les autres phases.

L'utilisation d'une méthode unique pour la conception formelle de systèmes informatiques à l'Institut, est aussi de nature à modifier profondément l'approche de l'outil informatique par les scientifiques, comme le démontre les premières expériences qui ont été tentées.

Elle devra offrir une alternative au mode de prototypage qui s'est généralisé jusqu'à maintenant, favorisé par la carence chronique d'informaticiens, et aider à la mise au point de logiciels spécialisés, dont l'utilisation systématique et rationnelle par

l'ensemble du personnel scientifique est indispensable pour garantir le développement de notre recherche scientifique.

INTRODUCTION

L'essor spectaculaire de l'informatique, principalement depuis l'apparition des microprocesseurs il n'y a guère plus de dix ans, a eu pour conséquence une diversification considérable de son champ d'application, de telle sorte que le logiciel est devenu un constituant d'un nombre de plus en plus important de produits industriels et/ou de services.

Un institut de recherche comme l'ORSTOM est somme toute une entreprise comme une autre, dont la finalité est de produire des connaissances et de l'information en général : comme telle, elle investit des moyens relativement importants dans des opérations de logiciels qui accompagnent ses actions scientifiques de génération de connaissances.

La production de logiciels est ainsi une activité qui doit se développer considérablement dans les années qui viennent à l'Institut, compte tenu de l'étendue des disciplines couvertes et de leur spécificité d'une part, des conditions particulières d'intervention et de réalisation de ces activités d'autre part.

Au fur et à mesure que l'outil informatique se généralise et atteint progressivement tous nos secteurs d'activité, les logiciels à élaborer deviennent de plus en plus importants, non seulement parce que l'on attaque naturellement des tâches à automatiser de plus en plus complexes, mais aussi parce que les utilisateurs potentiels, mieux informés, deviennent de plus en plus exigeants sur la fiabilité et la convivialité des produits qui leur sont destinés.

En se restreignant à des activités purement scientifiques, on peut classer *grosso modo* les logiciels à développer en deux grands pôles : les produits destinés à la capitalisation d'information (génération de bases de données) sans privilégier une application particulière de façon à n'en compromettre aucune, et ceux dont la finalité est l'exploitation des données dans l'optique d'interprétation de l'information correspondante. A priori, la première catégorie requiert normalement des services plus professionnels que la seconde, eu égard essentiellement à son caractère plus général.

L'introduction de l'informatique pour une activité déterminée à l'ORSTOM dépend normalement de l'initiative d'un chercheur et, en conséquence, du niveau de connaissances que celui-ci acquiert sur les techniques de traitement de données disponibles et applicables dans le domaine thématique qui l'intéresse. Cette phase constitue donc une première étape, dont l'acquis principal est de montrer la voie en mettant en évidence les données susceptibles de

faire l'objet d'un traitement informatisé et les fonctions qui peuvent être automatisées. Elle se caractérise par une faible productivité due à des tâtonnements et à une évolution progressive du logiciel produit, de toutes façons très personnalisé (il est notoire qu'un logiciel réalisé dans ces conditions survit rarement au départ de celui qui l'a conçu).

Cette évolution s'explique simplement par le fait qu'une fois satisfaits les besoins fondamentaux des utilisateurs, leurs imaginations travaillent et définissent de nouvelles nécessités. Cette phase peut donc être assimilée à une étape d'élaboration de prototypes successifs, pour laquelle l'utilisation de langages de quatrième génération est adéquate, car plus efficace, plus dynamique et partant mieux appropriée que le passage par une spécification formelle, qui conduit à des systèmes beaucoup plus longs à mettre au point.

L'examen de quelques expériences de développement de logiciels, déjà conclues ou encore en cours à l'ORSTOM, dénote une très faible productivité globale : il est bien sûr difficile de la chiffrer et de se livrer à des comparaisons, eu égard à la grande diversité des applications développées. On sait néanmoins que dans la plupart des cas un système requiert plusieurs années pour sa mise au point.

Par ailleurs, au moment où "*qualité*" est devenu un mot d'ordre pour toute activité économique, la qualité du logiciel doit retenir l'attention des concepteurs et projecttistes.

L'amélioration de la productivité dans la mise au point de logiciels, le souci de la qualité et le besoin de maîtriser la complexité sans cesse croissante des systèmes à développer exigent que l'ORSTOM se donne les moyens d'acquérir le métier du développement de logiciel. Ce savoir-faire se caractérise essentiellement par l'adoption et l'application de méthodes pour la réalisation de toutes les activités nécessaires à l'élaboration de "*produit-logiciels*".

La figure 1 présente les différentes étapes du développement d'un logiciel, selon la vision de Tom de Marco, l'un des fondateurs de l'analyse structurée (De Marco, T. 1978). Cet article a pour objectif de présenter une méthodologie susceptible d'être appliquée dans l'étape de conception d'un logiciel et les bénéfices que l'on peut espérer tirer de son adoption. Adaptée à partir des concepts avancés par l'école nord-américaine (Gane, C. et Sarson, T. 1979), cette méthode a été appliquée depuis deux ans par l'équipe franco-brésilienne du programme SISGEO pour la réalisation de logiciels dans des domaines aussi divers que climat, sols, végétation, documents cartographiques, images de satellites, etc..

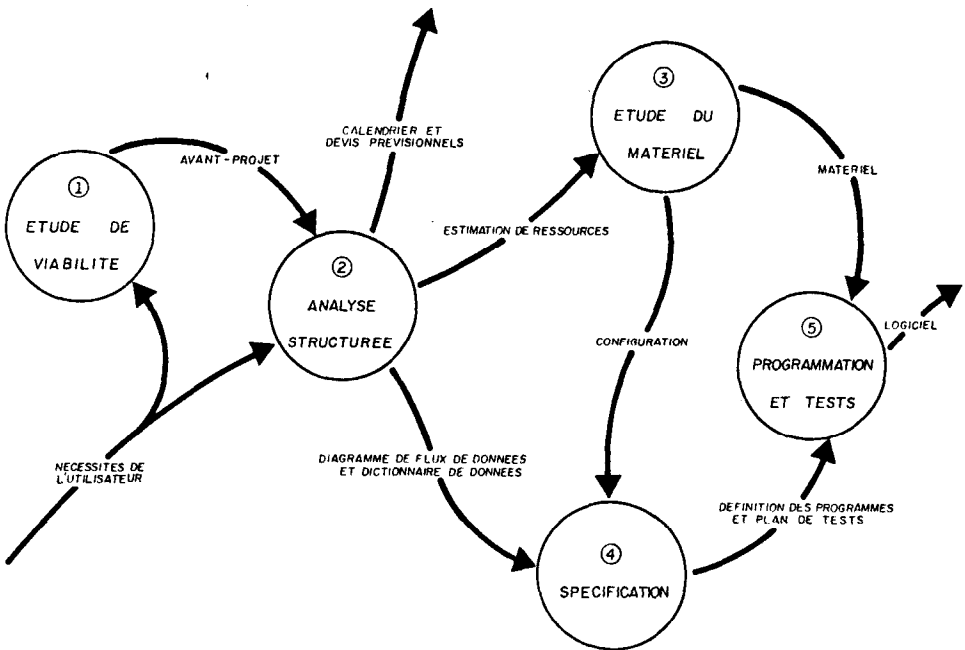


Fig. 1 - Etapes du développement d'un logiciel, vues par Tom de Marco (1978)

1. AVANTAGES

La définition et l'emploi d'une méthodologie répond à l'objectif de rendre plus homogène, donc plus efficace et rationnel, le développement des applications informatiques quel qu'elles soient. En ce qui concerne la seule partie de conception, l'adoption d'un ensemble de méthodes, normes et conventions répond aux préoccupations suivantes :

- *permettre le dialogue entre l'analyste et l'utilisateur.* Le succès d'une réalisation informatique est étroitement lié au degré d'intervention de l'utilisateur, à qui elle est destinée, dans son développement. Par conséquent, le souci majeur de la méthodologie de conception doit être de favoriser la compréhension qu'a un utilisateur des spécifications fonctionnelles établies par l'analyste. Elle doit en effet fournir un cadre apte à *définir clairement et enregistrer les nécessités informatiques de l'utilisateur.* On peut observer que dans la situation actuelle à l'ORSTOM, l'accès à l'informatique est réservé à des chercheurs qui ont consenti un effort particulier dans ce domaine : de ce fait, nombre de systèmes importants ne peuvent être mis au point, faute d'une connaissance technique suffisante des chercheurs intéressés. Dans la mesure où l'on dispose de quelques professionnels maîtrisant les techniques de développement de logiciel, l'adoption d'une méthodologie pour la spécification fonctionnelle devra *aider les scientifiques non initiés à l'informatique à accéder aux logiciels spécifiques dont ils ont besoin*, en jouant le rôle de responsable scientifique de l'application à mettre au point ;

- *permettre la documentation de l'application durant le développement.* Traditionnellement, la documentation de systèmes constitue le cauchemar de nos développeurs, de telle sorte que l'absence d'une documentation adéquate reste l'obstacle majeur à la valorisation des logiciels produits à l'ORSTOM. La plupart du temps elle est totalement inexistante ou, ce qui n'est guère mieux, elle se résume à l'insertion de quelques commentaires dans des programmes Basic ou Fortran, restreignant automatiquement le public atteint aux programmeurs dans ces langages. Dans les meilleurs cas, c'est à dire limités à une poignée de logiciels produits à l'Institut, un manuel d'utilisation est rédigé à posteriori, sans autre règle pré-établie que l'analogie avec les exemples de manuels qui accompagnent les logiciels du commerce. Il convient de noter que cet effort, louable au demeurant, reste incomplet dans la mesure où ce manuel ne sera d'aucun secours pour une éventuelle maintenance (évolution) postérieure. De la méthode d'analyse fonctionnelle résulte un ensemble de documents réunis dans le projet logique, qui constitue la base de la documentation du système : son

grand avantage est d'être produit à un moment où il peut servir de référence et de point de départ pour les phases suivantes (figure 1) ;

- *permettre le dialogue entre l'analyste et le programmeur.* En l'absence de tout document de spécification fonctionnelle ou organique, l'analyste est condamné à devenir à la fois le projet-tiste et le programmeur, ce qui l'oblige à assumer l'intégralité du développement du système. Cette situation s'est beaucoup répétée par le passé, de telle sorte que bon nombre d'analystes sont devenus esclaves de leur produit ("*analyste d'un seul système*"), faute d'avoir su partager les diverses tâches. Avec la complexité croissante des applications à développer, dans la plupart des cas on ne peut plus concevoir qu'un logiciel soit développé par une seule personne, de telle façon qu'une méthode rigoureuse est indispensable pour transmettre l'étape de conception (quelles fonctions seront automatisées ?) à l'étape de spécification (comment le seront-elles ?) ;

- *retarder au maximum la phase de programmation.* Tant que durent les premières phases de développement d'un logiciel (analyse fonctionnelle et analyse organique), l'impact d'une modification du projet est moins sensible, dans la mesure où aucune ligne de code n'a encore été écrite. Les échanges entre l'analyste et l'utilisateur pendant la phase de conception, puis entre ce dernier et le projet-tiste durant la phase de spécification, permettent d'examiner "*en profondeur*" les choix faits par les professionnels de l'informatique dans le but de satisfaire aux exigences du système. L'éventualité d'avoir une transformation importante à réaliser dans les phases postérieures du développement en est donc d'autant moins probable. Dans cette optique il est recommandable de ne pas commencer la phase de spécification tant que le projet logique n'a pas été dûment approuvé par toutes les personnes concernées. Avec cette méthode, le nombre d'erreurs fondamentales de programmation tend à diminuer sensiblement, ce qui compense normalement le temps passé aux spécifications ;

- d'une façon tout à fait semblable, le fait de suivre une méthodologie bien définie pour la conception du logiciel devra *permettre l'acquisition d'une configuration matérielle basée sur des critères rationnels et objectifs*, évitant ainsi toute précipitation pour l'achat d'un équipement, situation que l'on rencontre couramment ;

- *permettre de définir clairement ce qu'un logiciel fera.* Le principal objectif du projet logique est d'identifier les fonctions et processus que le logiciel à développer devra automatiser. Il est alors facile de savoir ce que le logiciel fera, et ce qu'il ne fera pas, et selon quelle logique, cette dernière étant également dans l'étape de conception par l'analyste ;

- la décomposition du logiciel en diverses fonctions ayant des objectifs distincts *permet le développement modulaire d'un système tout en conservant la vision d'ensemble* fournie par le projet logique. En effet, la complexité des systèmes constitue souvent un facteur qui empêche, ou pour le moins freine, leur réalisation : la structuration en processus, fréquemment indépendants, qui est réalisée au cours de l'étape de conception permet d'envisager une mise au point progressive, fonction après fonction, pouvant conduire à des versions successives de plus en plus élaborées et complètes du logiciel ;

- *fournir un argument pour suspendre toute modification pendant la réalisation.* Il est en effet souvent difficile de refuser une modification demandée par l'utilisateur pendant la réalisation du logiciel, même si celle-ci remet en question certains des choix déjà réalisés. Une fois le projet logique adopté, on peut renvoyer toute modification structurelle à une version postérieure du système ;

- *standardiser la conception des logiciels et en favoriser la valorisation.* Dans la mesure où l'on adopte une méthodologie et où l'on s'astreint à produire un document obéissant à des règles pré-établies, on permet une dépersonnalisation des systèmes élaborés et une normalisation favorable à la valorisation des produits mis au point. Dans cette optique, un manuel de normes et règles pourrait être élaboré de manière à *fournir une méthodologie pour la documentation des logiciels*, indispensable à leur institutionnalisation ;

- *produire des systèmes moins dépendants d'une réalisation physique.* Cet aspect constitue certainement un des éléments les plus positifs apporté par l'adoption d'une méthode de conception spécifique. Pour accompagner la dynamique actuelle de l'informatique, la plupart des applications développées ont eu à subir un certain nombre de migrations, de manière à s'adapter à de nouvelles configurations matérielles. Dans ces conditions, le projet logique établi à l'issue de la phase de conception est conservé, dans la mesure où il ne fait aucune référence à une configuration matérielle déterminée : il peut ainsi être regardé comme le document fondamental de l'application ;

- *fournir une voie pour l'utilisation de techniques de génie logiciel.* A l'heure actuelle le génie logiciel constitue, avec l'intelligence artificielle, l'un des domaines les plus effervescents de la recherche en informatique. Son but est d'apporter un soutien aux activités de développement de systèmes, en automatisant un maximum des tâches afférentes à cette activité. Un certain nombre de produits commerciaux sont déjà disponibles, dans ce domaine : aide à la conception, aide à la spécification, aide à la programmation ; tous ont pour base une méthodologie théorique existante (analyse structurée et variantes, projet structuré et variantes, etc.).

L'adoption d'une méthode est donc un préalable indispensable à l'utilisation des techniques de génie logiciel, susceptibles d'apporter un gain de productivité et de qualité significatifs lors de la mise au point des applications informatiques (Bourgeois, J. 1984).

D'autres avantages secondaires devraient également résulter de l'adoption d'une méthode de conception : certains sont évidents, comme le fait de *faciliter le dialogue entre les développeurs*, en leur permettant une communication plus technique grâce au langage commun qui est créé. Malgré la grande diversité des thèmes abordés à l'ORSTOM, chacun pourra facilement profiter de l'expérience de ses collègues, et faire bénéficier ceux-ci de sa propre expérience. D'autres avantages sont plus subtils, comme celui *d'autoriser un suivi de l'état d'avancement du projet*, alors que, traditionnellement, il est toujours très difficile de savoir où en est la mise au point d'une application, ni même d'avoir une prévision raisonnable de la conclusion d'une étape déterminée.

En se limitant à la seule phase de conception, on retiendra utilement tout l'intérêt que l'on peut tirer de l'élaboration d'un projet logique qui s'applique à définir exactement ce que le système fera, en faisant totale abstraction de toutes préoccupations liées à l'implémentation physique (restrictions de matériel, organisation physique de l'information, conditions d'accès, etc.), et sans compromettre les options qui pourront être prises ultérieurement en accord avec l'environnement matériel choisi ou disponible.

2. METHODE D'ANALYSE STRUCTUREE

2.1. La philosophie

La méthode de conception de systèmes qui est résumée dans cet article est adaptée de la version due à Chris Gane et Trish Sarson, qui a été publiée en 1979 dans l'ouvrage "*Structured System Analysis : tools and techniques*". Celle-ci fait elle-même partie d'une école de méthodologies semblables (Yourdon, E. & Constantine, L.L., 1979), (De Marco, T., 1978), (Warnier, D.J. 1985), etc..

L'idée essentielle est de permettre la représentation graphique des fonctions d'un logiciel, et des données sur lesquelles ces fonctions agissent, de façon suffisamment simple et claire pour servir de véhicule de communication entre toutes les parties intéressées par la réalisation d'un système. Le diagramme correspondant peut alors être considéré comme un modèle logique du système à développer.

De cette façon, le problème principal (absence de langage commun entre l'analyste et l'utilisateur) est résolu, sans que l'analyste ait besoin de tout connaître de la spécialité dont relève le logiciel (il n'aura à poser des questions que sur les informations qui

doivent être manipulées, et qui sont représentées sur le diagramme), et sans que l'utilisateur n'ait plus à apprendre de l'informatique que les quelques symboles utilisés dans cette représentation graphique.

Une autre particularité de ces diagrammes est qu'ils permettent une approche descendante "*top-down*" du système, en autorisant une décomposition successive de chaque fonction mise en évidence dans une première représentation, pour fournir un nouveau diagramme de niveau plus détaillé.

Une fois les diagrammes élaborés, les entrées-sorties de données qu'ils mettent en oeuvre sont décrites et détaillées, de manière à construire le "*dictionnaire de données*". Celui-ci constitue le second produit essentiel de l'étape de conception : il s'agit d'une structure adéquate pour recevoir tout le détail descriptif des données et de la logique des fonctions qui les transforment.

2.2. Les diagrammes de flux de données

Le diagramme de flux de données (DFD) est une maquette du système à développer : il doit donc représenter, en plus des données concernées (*flux*, ou flots) et des transformations qui leur sont appliquées (fonctions et processus), les personnes qui sont à l'origine, ou à qui sont destinées ces informations : ce sont les entités externes. Bien entendu, lors de l'analyse on est amené à identifier des étapes entre les processus pour lesquelles il est nécessaire de définir le stockage de certains éléments de données : le terme "*dépôt de données*" est utilisé pour représenter le concept logique de lieu de stockage.

Par conséquent, au total quatre symboles sont nécessaires pour construire le modèle logique du système (figure 2a). Un carré, dont les deux côtés en haut et à gauche sont représentés par des traits épais, symbolise une entité externe. Les flux de données sont représentés au moyen d'une flèche, de préférence horizontale ou verticale, dont la pointe indique la direction du flux. Les fonctions figurent sous la forme d'un rectangle debout, avec les coins arrondis : elles sont identifiées par un numéro situé dans la partie supérieure du rectangle. Enfin, le dépôt de données est symbolisé par une paire de lignes parallèles horizontales reliées à une seule des extrémités, et est identifié par un D majuscule suivi d'un numéro de séquence.

Chaque symbole sur le diagramme de flux de données reçoit un nom et une description, lesquels doivent être choisis de façon à être suggestifs pour l'utilisateur :



Fig. 2a - Les diagrammes de flux de données sont construits avec seulement quatre types de symboles

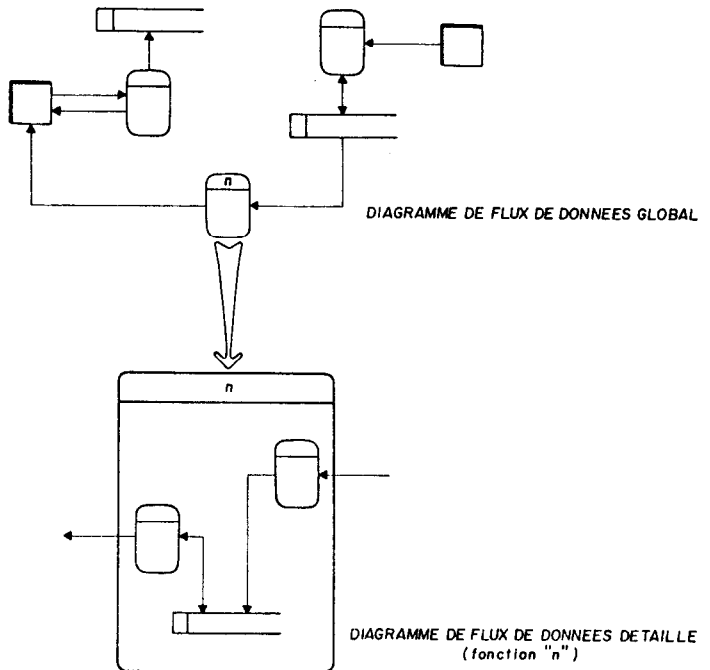


Fig. 2b - Chaque fonction du diagramme de flux de données global peut être "explosée" en un nouveau DFD

- *l'entité externe* est normalement une catégorie de personnes qui se communiquent avec le système : on utilise alors un terme générique (Hydrologue, Pédologue, par exemple). Il peut aussi s'agir d'un autre système qui reçoit ou fournit des informations au logiciel en cours d'analyse ;

- *le flux de données* doit être identifié de forme globale par une description qui caractérise le mieux possible son contenu. Par exemple : Sollicitation de calculs, Détails de jaugeages, Données d'identification, Caractéristiques de transaction, etc.. En règle générale, un même flux de données peut intervenir plusieurs fois sur les DFD, par exemple à l'entrée d'une fonction "Mettre à jour" et à la sortie de cette même fonction, en ayant le dépôt de données comme destination. On conviendra alors de ne faire figurer qu'une seule fois sa définition, de manière à éviter le risque de répétition dans la phase ultérieure d'analyse de données (Cf. 2.3) ;

- *la fonction* reçoit une description résumée qui se présente sous la forme d'une sentence impérative : verbe à l'infinitif + objet. Par exemple : Calculer les débits journaliers, Mettre à jour les caractéristiques, Emettre bulletin mensuel, Générer vecteur régional. On évitera les verbes comme "traiter" ou "réviser" qui montrent une connaissance encore insuffisante de la fonction en question, ce qui laisse supposer qu'une nouvelle décomposition est nécessaire. Par ailleurs, "ordonner", "classer" ou "trier" doivent également être évités, car ils représentent des fonctions sans valeur logique (simple réorganisation physique d'un fichier, qui n'a d'objet qu'après un choix de séquence ou d'organisation). La même réflexion s'applique pour les processus de simples copies et récupérations de sécurité ("*back-up/restore*"). Certaines des fonctions mises en évidence sur le diagramme peuvent être décrites comme "manuelles", lorsque le processus exercé (type Préparer les données, Extraire, etc.) est mal défini et, partant, non passible d'une automatisation, du moins dans l'état actuel des connaissances que l'on en a. On préférera toutefois remplacer ce type de fonction si possible par une entité externe (type : Coordinateur de transcription, Préparateur de cartes à digitaliser, etc.) ;

- *le dépôt de données* est identifié simplement par le nom qui caractérise son contenu : Tables, Etalonnages, herbier, Unités cartographiques, etc.. On évitera les termes inutiles comme Fichier de .., Cadastre, (Tables) du système.

Pour ne pas compliquer le diagramme de flux de données avec des lignes qui se croisent, la même entité externe ou le même dépôt de données peuvent être dessinés plus d'une fois sur le même diagramme : les deux (ou plus) symboles par entité externe sont identifiés par une ligne inclinée dans le coin inférieur droit du carré, tandis que les deux (ou plus) symboles par dépôt de données reçoivent une ligne verticale additionnelle à gauche.

Lors de l'élaboration d'un diagramme de flux de données initial, l'analyste doit se préoccuper de montrer le cadre général du système : il est donc indispensable de renvoyer la description de certains détails (conditions d'erreur ou d'exception, par exemple) à un niveau inférieur. De cette façon, chacune des fonctions illustrées sur le DFD de niveau 0 (encore appelé diagramme de contexte) peut être explosée sur un diagramme de flux de données de niveau inférieur (niveau 1), plus détaillé (figure 2b).

Si nécessaire chaque processus détecté à la suite de la décomposition d'une fonction peut être de nouveau subdivisé pour présenter un niveau supérieur de détails. Cette décomposition n'est toutefois généralement pas indispensable : il existe un certain nombre de règles, tant pour l'élaboration des DFD que pour en traiter les explosions successives.

2.3. Le dictionnaire de données

A mesure que l'on augmente les détails de l'analyse fonctionnelle, qui a pour résultat les DFD, on est amené à préciser la logique des fonctions, les détails des flux de données, le contenu des dépôts de données, etc.. Le dictionnaire des données constitue la structure adéquate pour recevoir les détails descriptifs des éléments des diagrammes de flux de données. Il est important d'observer que pendant toute cette phase on continue à se maintenir au niveau logique, en évitant de dessiner un format de bordereau, d'esquisser un gabarit de rapport d'éditer un "lay-out" de fichier!

Le dictionnaire des données du système peut être élaboré dès que les DFD sont considérés comme satisfaisants : l'étape correspondante est appelée analyse des données, et consiste à reprendre chaque diagramme et décrire successivement tous les symboles rencontrés dans la représentation graphique. On est alors conduit à définir avec précision :

- *les entités externes*. Il y a normalement peu d'entités externes pour un logiciel donné : en général on pourra se satisfaire de la liste de leur noms (tels qu'ils apparaissent sur les DFD) et de la description correspondante (normalement donnée par l'utilisateur), classée par ordre alphabétique. Dans le cas où une entité externe représente un autre système, il pourra être nécessaire de préciser son environnement sur le plan opérationnel et s'il existe un processus d'interface déjà prévu ;

- *les fonctions* sont classées conformément à leur numéro d'identification sur les DFD. En plus du numéro d'identification et de la sentence qui définit la fonction sur le diagramme, le dictionnaire de données devra contenir, pour chaque fonction, une description succincte et une spécification, aussi précise que possible, de la logique interne du processus. Celle-ci

pourra être exprimée sous la forme de langage structuré, tables ou arbres de décision. Pour éviter une description logique très importante dans le cas des fonctions les plus complexes, on pourra faire référence à des algorithmes pour lesquels un sous-chapitre spécial est ouvert dans le projet logique. Enfin, on conviendra de placer systématiquement, comme première clause de la description fonctionnelle, la nécessité éventuelle d'exécuter antérieurement une autre fonction, de façon à traduire l'existence d'une séquence logique entre deux processus ;

- *les flux de données*. Il s'agit de structures de données possédant une origine et une destination. Dans la mesure où un flux entre ou sort par rapport à une fonction, on lui attribue le code de la fonction suivi de lettre E ou S. En effet, il ne peut y avoir de flux de données entre deux entités externes, entre deux dépôts de données ou entre une entité externe et un dépôt de données sans intervention d'une fonction. Il est important de noter dans le dictionnaire, pour chaque flux, sa périodicité et une estimation du volume qui intervient à chaque exécution de la fonction, de façon à fournir une orientation utile pour la spécification et le projet physique. La description du contenu du flux de données est réalisée en détaillant la liste des éléments de données qui composent ce flux, chaque élément de données étant lui-même défini dans le dictionnaire de données sous la rubrique "*champs*".

Tous les *champs* qui apparaissent dans les flux de données, mais aussi dans l'expression de la logique des fonctions doivent être décrits : toutefois, les champs de contrôle, c'est à dire ceux qui sont seulement utilisés pour les nécessités internes du système n'y figurent pas. Il est intéressant de prévoir un formulaire spécifique pour recevoir les nombreuses caractéristiques de chaque champ, de façon à faciliter l'élaboration du dictionnaire de variables. Ces caractéristiques peuvent être regroupées selon les catégories suivantes : identification, pseudonymes, structure, origine, stockage, domaine de valeurs, taille et représentation, codification, validation.

L'identification d'un champ est constituée par un *symbole* mnémorique, choisit suffisamment court pour ne pas compromettre l'utilisation de tel ou tel langage au moment de la programmation (huit lettres pourrait être un maximum). Le choix de ce symbole en temps de conception, et son appartenance au dictionnaire de variables est une garantie évidente d'homogénéité postérieure.

La *description*, aussi précise que possible, est normalement obtenue auprès de l'utilisateur, ainsi que le *nom d'usage courant*, optionnel, destiné à conserver comme pseudonyme la désignation usuelle traditionnelle dans la discipline (par exemple : TX et TM pour les températures maximales et minimales, sont des symboles consacrés par les climatologistes).

La *structure* d'un champ est destinée à maintenir, dans le dictionnaire de variables, une information sur les champs composés et la relation entre les divers éléments d'une même structure. Elle pourra être représentée par un ensemble comme XX.Y, où XX représente le numéro de séquence d'un champ composé dans un groupe (flux ou dépôt) et Y le numéro de séquence d'un sous-champ dans ce champ. Par exemple :

DATE	01
JOUR	01.1
MOIS	01.2
ANNEE	01.3

L'*origine* d'un champ peut être spécifiée, à ce stade du développement, par le code du flux de données, ou de l'algorithme, qui lui donne naissance. Toutefois, dans la phase postérieure de spécification, il sera plus intéressant d'indiquer le code du formulaire (ou de l'écran) pour les champs qui proviennent d'un flux de données.

Le *stockage* est noté à partir du code d'identification du dépôt de données qui contient ce champ : de la même façon, en temps de projet physique, on pourra indiquer le code mnémonique du fichier correspondant.

Comme un même champ peut figurer dans plusieurs flux et être contenu dans plus d'un dépôt de données, il est indispensable de prévoir plusieurs occurrences, tant pour l'origine que pour le stockage.

Le domaine des valeurs possibles est implicitement défini par le *type* du champ, numérique ou alphanumérique. Un *intervalle de valeurs* peut normalement être donné pour les champs numériques : le fait de préciser les valeurs minimale et maximale d'une variable en permettra une vérification plus rigoureuse. Dans certains cas il est nécessaire de préciser une valeur (ou un état) par *défaut*. Bien entendu, l'*unité de mesure* doit également être notée en temps de projet logique, en accord avec les préférences de l'utilisateur : dans le cas où le logiciel prévoit la conversion d'unités de mesure pour une variable déterminée, on utilisera deux champs différents.

La taille d'un champ dont on connaît déjà le type sera simplement représentée, sans faire référence au matériel, par un *masque d'édition*.

Dans le cas où le champ possède plusieurs *occurrences* (tableau), on en notera le nombre pendant la phase de conception.

La codification d'un champ sera représentée simplement dans le dictionnaire de variables par le numéro d'identification de la *table* utilisée pour décodifier le champ en question.

La validation d'un champ fait l'objet d'une attention particulière lors du montage du dictionnaire de variables. Elle peut s'exprimer par l'ensemble des caractéristiques suivantes :

- *obligatoire*. Il s'agit d'un booléen qui signale si le champ correspondant doit être présent impérativement, ou non ;

- *validation spéciale*. Elle s'exprime sous la forme du code de l'algorithme de validation de ce champ. Par exemple, pour une longitude, le sous-champ hémisphère ne peut être que Est ou Ouest ;

- *cohérence*. Elle indique l'existence d'un algorithme que la valeur de ce champ doit respecter, en combinaison avec un ou plusieurs autres champs. Par exemple, en chimie des sols les divers résultats de l'analyse granulométrique de la terre fine (sable grossier, sable fin, silte et argile) doivent totaliser 100% pour un échantillon déterminé. Le code de l'algorithme en question devra bien sûr figurer pour chacun des champs qui y interviennent ;

- *erreur*. Cette colonne est utilisée pour signaler le cas d'un champ qui devra être accepté, même s'il a été considéré erroné par la critique. Ce cas intervient fréquemment pour des données historiques (climatologie, par exemple), lorsqu'il n'y a pas de possibilité de retrouver la valeur réelle, l'erreur étant alors tout au plus signalée ;

- *digit de vérification*. On signale l'existence d'un digit de contrôle pour le champ considéré.

Dans le sous-chapitre *algorithmes* sont regroupées toutes les formules de calculs mathématiques (généralement utilisées pour obtenir un élément de donnée en fonction d'un ou plusieurs autres), ainsi que les expressions en langage structuré et autres tables de décision, utilisées dans le système. Chaque algorithme reçoit un code (A, suivi d'un numéro de séquence à deux chiffres), une définition et la description mathématique de sa logique. Pour chaque algorithme il est essentiel de traiter tous les cas d'exception avec l'utilisateur, rencontrés lorsque l'un ou l'autre des arguments est inconnu (par exemple).

Les *dépôts de données*, enfin, constituent également des structures de données : ces structures sont cette fois statiques, alors que les flux de données étaient des structures dynamiques. Le contenu d'un dépôt de données sera donc décrit en fournissant la relation des noms de champs qui composent cette structure.

On précisera de plus le code et le nom du dépôt de données, ainsi qu'une description succincte de son utilité. A ce stade de la conception on pourra faire une observation sur la clé primaire, voire sur l'organisation physique : toutefois, la clé peut ne pas être unique dans la mesure où aucune normalisation n'a encore été faite.

Pour terminer, il convient de faire trois remarques importantes au sujet du dictionnaire de données :

- en premier lieu, en suivant scrupuleusement les règles dictées pour son élaboration, on garantit que tous les éléments nécessaires à la réalisation du système sont décrits, et en un seul endroit. La logique est trouvée dans la partie des fonctions, les entrées-sorties dans la partie sur les flux de données, les caractéristiques des données dans la partie des champs et les nécessités de stockage dans la partie sur les dépôts de données. Le projetiste va donc pouvoir y rencontrer toutes les spécifications dont il a besoin pour dessiner les fichiers, créer les écrans, élaborer les gabarits des états, etc., qui constituent les tâches du projet physique ;

- on a pris le plus grand soin, dans toute l'analyse des données, de ne jamais aborder le moindre aspect de la réalisation concrète, de manière à ne pas (trop) induire une solution physique particulière. Par conséquent, le projetiste reste entièrement libre de ses choix quant à l'organisation des fichiers, la structuration des tâches, les supports pour les entrées-sorties, etc.. De plus, la partie projet logique reste totalement indépendante de la version implémentée qui en sera faite dans un environnement matériel déterminé ;

- la partie dictionnaire de variables proprement dite peut être automatisée, compte tenu du nombre élevé de variables que certains systèmes peuvent atteindre (cent variables est un nombre très fréquent, cinq cents n'est pas si rare), et surtout de l'intérêt que l'on peut en tirer. En effet, on peut imaginer de nombreuses applications de ce dictionnaire : rapports totaux ou partiels, complets ou résumés, selon diverses classifications ; vérifications de cohérence et intégrité ; génération automatique de "*books*" ou "*includes*", etc.. On notera cependant que l'essentiel de la valeur d'un dictionnaire de variables provient de ce qu'il constitue un dépôt central d'information sur le système, pour l'analyste, le projetiste et les programmeurs qui y travaillent. Ce dernier élément doit

être pris en considération lors du choix éventuel d'un logiciel d'administration de dictionnaire, en particulier en ce qui concerne la configuration le supportant (ressource distribuée ou non).

2.4. Le reste du projet logique

Pour compléter la documentation de l'étape de conception, un certain nombre d'éléments doivent être précisés, et font l'objet d'autant de chapitres du projet logique correspondant. En prenant exemple sur la méthodologie adoptée par la division de développement de systèmes de l'EMBRAPA, et appliquée par l'équipe de SISGEO, le projet logique pourrait avoir le sommaire suivant :

1. Introduction
2. Modèle d'entités/rerelations
3. Diagrammes de flux de données
4. Dictionnaire de données
5. Tactique de réalisation
6. Estimation de ressources
7. Glossaire
8. Signatures

Le premier chapitre, introduction, a pour objet de décrire succinctement le contexte général de l'application, essentiellement en termes de présentation de ce qui motive son développement. Les sous-chapitres suivants constituent cette introduction : historique (brève narration des faits qui ont conduit à la conception du système), objectifs, en termes généraux, du système et (justificatifs)/bénéfices attendus de sa mise au point.

Le chapitre suivant, modèles d'entités/rerelations, est constitué d'un diagramme schématisant ce modèle et de la liste alphabétique des entités (nom, description et identification du dépôt de données correspondant). Ce chapitre est normalement rédigé à la fin de la phase de conception, car il repose sur la normalisation des dépôts de données qui interviennent dans le logiciel, jusqu'à la troisième forme normale.

La première forme normale correspond à l'élimination des groupes répétés dans une structure de données déterminée. La deuxième correspond au regroupement de tous les champs qui dépendent d'une même clé tandis que la troisième forme normale suppose l'élimination des dépendances fonctionnelles entre champs distincts de la clé primaire, ce qui revient à isoler les tables de conversion qui expriment ces dépendances.

Le modèle d'entités/rerelations constitue une vision logique qui se traduit dans le projet physique par le *lay-out* de la base de données. Toutefois, ce dernier incorpore déjà certaines restrictions imposées par l'environnement de "*hardware et software*".

Les chapitres 3 et 4 ont été amplement discutés dans les parties précédentes de cet article (§2.2 et §2.3). Le cinquième chapitre, tactique de réalisation, a pour but de fournir un calendrier prévisionnel pour la mise en place effective du logiciel : l'élaboration d'un diagramme de structure du système, mettant en évidence les principales fonctions indépendantes (sous-systèmes), sera utile pour la compréhension de la décomposition en étapes distinctes. Le second sous-chapitre, sur les restrictions, doit présenter toutes les limitations susceptibles de freiner, voire empêcher, l'élaboration du logiciel, ou encore en restreindre l'utilisation. Par exemple, ce pourra être une indisponibilité de personnel, l'absence de matériel adéquat, des restrictions administratives ou d'accès aux informations, etc.. Le dernier sous-chapitre doit aborder le mode d'implémentation, en spécifiant essentiellement si la réalisation doit être modulaire, quelles sont les fonctions prioritaires, quel est le calendrier prévisionnel.

Le sixième chapitre, estimation de ressources, poursuit l'objectif de déterminer, dès la phase de conception, la demande de ressources nécessaires à la mise en oeuvre du système. Il contient donc un certain nombre d'éléments de nature à fixer la configuration minimale et la configuration recommandée pour le produit en question. Dans le cas d'un logiciel destiné à être exécuté sur un micro-ordinateur, les caractéristiques suivantes devraient y figurer : type de CPU, taille de mémoire vive utilisée, nécessité éventuelle d'un coprocesseur arithmétique, nombre de drives, type de carte vidéo, type d'impression (sur 80 ou 132 colonnes), nécessité de périphérique spécifique comme traceur de courbes ou table à digitaliser, capacité de stockage minimale (disquette ou disque dur), système d'exploitation (version minimale) et autres logiciels nécessaires à l'utilisation de ce produit.

De la même façon, une estimation des ressources humaines (temps d'opération mensuel, par exemple) et des ressources financières requises pour le fonctionnement normal du système devrait être fournie dans ce chapitre.

Le septième chapitre, glossaire, a pour but de définir clairement chacun des termes utilisés et qui n'apparaissent dans aucun des éléments décrits dans le dictionnaire de données. Il se compose normalement d'une partie thématique (termes introduits par le spécialiste) et d'une partie informatique (termes employés par l'analyste). La connaissance des expressions et des concepts contenus dans le glossaire enrichit la propre compréhension du système.

La dernière partie est destinée à recevoir la signature des personnes responsables pour l'approbation du projet logique. Il est en effet indispensable que ce document soit approuvé par les deux parties, et que chacun soit conscient de ce que la mise en route des étapes suivantes de spécification et implémentation signifie en

termes de bénéfices apportés par l'automatisation, mais aussi d'efforts, de ressources impliquées, de délais à attendre, etc.. C'est donc une sorte de contrat pour la mise au point du logiciel que l'utilisateur signe avec l'analyste responsable de la conception, au nom de la totalité de l'équipe de développement.

3. EXEMPLE D'APPLICATION

Pour illustrer la façon d'utiliser la méthode de conception qui a été présentée dans cet article, on s'appuiera sur la production récente d'un logiciel d'administration de dictionnaire, PCDICO.

3.1. Énoncé

L'énoncé du problème peut être extrait d'un rapport de mission effectuée en mars 1987 à Manaus, auprès de Françoise Grenand, linguiste, responsable scientifique du projet :

... Il s'agit de la manipulation sur micro-ordinateur d'un dictionnaire de plus de deux mille mots. Un premier système rudimentaire avait été réalisé avec le concours d'un programmeur brésilien sur micro-ordinateur CP/M, de telle sorte que le fichier a été entièrement saisi. Toutefois les ressources d'administration de données (tris, mise à jour, édition de champs) sont précaires et ne permettent pas de réaliser de manière satisfaisante les corrections nécessaires, d'autant plus que le fichier (1,2 Mégaoctets) doit être réparti sur 06 disquettes.

La migration de cette application sur PC-compatible, avec disque dur, a été décidée et, à cette occasion, une nouvelle version du système sera mise au point et comportera des ressources complètes d'édition de champ pour l'inclusion et la mise à jour, une structure et une organisation de fichier plus appropriées, des ressources de tri plus performantes. Le système correspondant devra être programmé à Brasília, et prêt pour la mi-mai 1987. ...

3.2. Diagrammes de flux de données

Les diagrammes de flux de données qui ont été établis sont particulièrement simples. Au niveau 0, diagramme de contexte de la figure 3a, deux fonctions sont représentées, l'une de mise à jour du dictionnaire accessible par le linguiste, l'autre des consultations accessibles par les utilisateurs.

Fig. 3a - Diagramme de contexte de PCIDICO

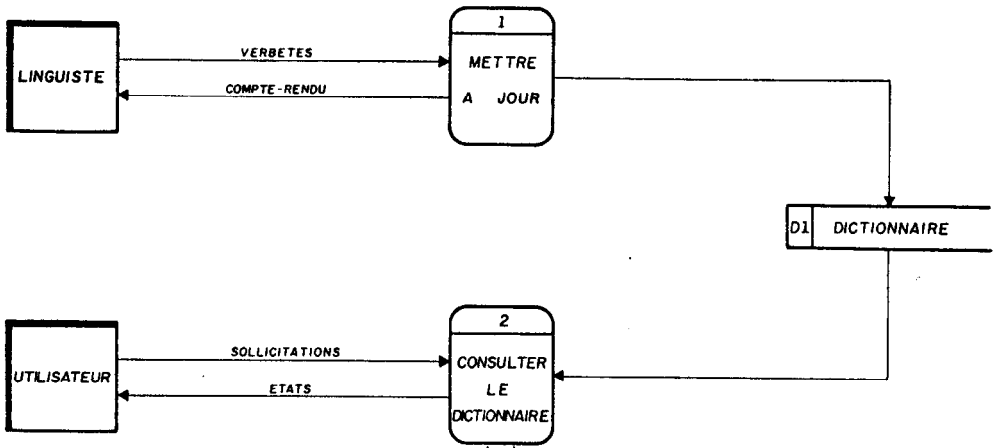
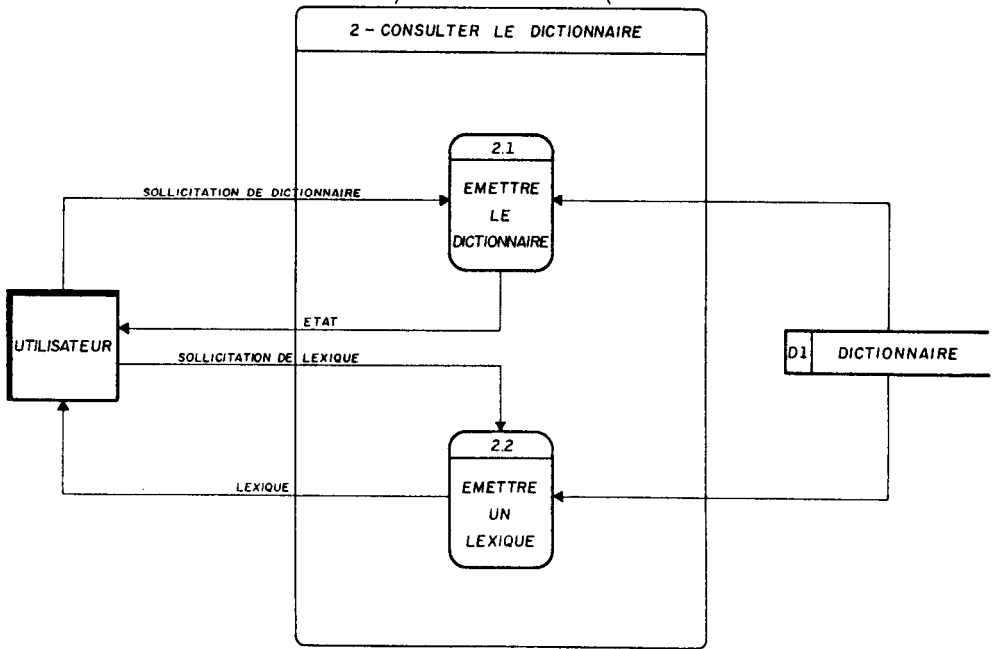


Fig. 3b - Détail de la fonction 2



A ce niveau, deux entités externes, un dépôt de données (le dictionnaire) et quatre flux de données sont mis en évidence. Pour faire apparaître deux types différents de sortie : état du dictionnaire (préparation à l'édition) et lexiques, la fonction "Consulter" a été éclatée et les flux de données correspondants précisés (figure 3b).

3.3. Dictionnaire des données

Le dictionnaire des données peut alors être élaboré.

a) définition des entités externes :

Linguiste - expert de la langue, responsable pour la constitution, l'administration et la maintenance du dictionnaire ;

Utilisateur - toute personne intéressée par la consultation et l'utilisation du dictionnaire, y compris l'éditeur et le propre linguiste.

b) définition des fonctions :

1 Mettre à jour : processus responsable pour la saisie, l'actualisation et la correction des données du dictionnaire. Celui-ci devra prendre en considération le problème de l'accentuation, c'est à dire manipuler les caractères accentués et le ç, représentés, par convention, sous la forme de deux caractères consécutifs (par exemple : 'e = é), de façon à conserver un ordre lexicographique plus naturel que l'ordre ASCII (ç juste après le c, en particulier).

2 Consulter le Dictionnaire : plusieurs types de consultation du dictionnaire seront autorisés : liste complète, liste spéciale pour l'édition, liste sélective des mots du dictionnaire, lexiques. Le résultat pourra être transmis sur écran ou sur papier.

2.1 Emettre le dictionnaire : production d'une liste des "verbetes" complètes, comme document préparatoire à l'édition. La sélection des verbetes selon divers critères est autorisée, pour permettre des recherches spécifiques.

2.2 Emettre un lexique : production d'un état simplifié, du type mot-sens (ou sens-mot).

c) définition des flux de données :

Verbetes (1.E) : ce sont les données qui altéreront le dictionnaire. Elles sont organisées en "verbetes".

- volume : il existe près de 2200 verbetes pour le "*pequeno dicionario de lingua geral*" ;

- périodicité : éventuelle ;

- champs : (attributs de verbete)
 - Mot (vocable de la "Lingua Geral")
 - ClasseMot (classe grammaticale du mot)
 - Sens (en portugais, signification du mot)
 - ClasseSens (classe grammaticale du sens)
 - Exemple (exemple d'utilisation du mot)
 - Etymologie (texte sur l'origine du mot)

Compte-rendu (1.S) : il s'agit d'un message sur une grille d'écran, qui confirme l'action effectivement réalisée sur le dictionnaire pour une transaction déterminée.

- volume : une grille d'écran par verbete altérée, insérée ou exclue ;
- périodicité : éventuelle ;
- champs : données de la verbete (voir 1.E) et message explicatif.

Sollicitation de dictionnaire (2.1E) : elle est constituée par une grille d'écran qui précisera les critères utilisés pour l'édition du dictionnaire (forme, support, sélection).

- volume : une grille d'écran par sollicitation ;
- périodicité : éventuelle ;
- champs : (diverses options pour l'émission)
 - Forme (complet ou simplifié)
 - Critère (champ composé)
 - Condition de préfixe
 - Condition de suffixe
 - Condition de classe grammaticale
 - Condition de mot
 - Condition de longueur (de mot)
 - Périphérique (écran ou imprimante).

Etat du dictionnaire (2.1S) : écran ou sortie d'imprimante contenant le résultat du processus d'édition d'états.

- volume : d'une page (ou un écran) à 250 pages (ou 500 écrans) par sollicitation. Dépend de la demande et du contenu actuel du dictionnaire ;
- périodicité : éventuelle ;
- champs : attributs de la verbete (voir 1.E) et message donnant les statistiques de l'édition.

Sollicitation de lexique (2.2E) : elle est constituée par une grille d'écran qui précisera les critères utilisés pour l'édition du lexique.

- volume : une grille d'écran par sollicitation ;
- périodicité : éventuelle ;
- champs : (diverses options pour l'émission)
 - Forme (complet ou simplifié)
 - Périphérique (écran ou imprimante).
 - Classement (lingua geral ou portugais)

Lexique (2.2S) : écran ou sortie d'imprimante contenant le résultat du processus d'édition de lexiques.

- volume : d'une page (ou un écran) à 70 pages (ou 200 écrans) par sollicitation. Dépend de la sollicitation et du contenu actuel du dictionnaire ;
- périodicité : éventuelle ;
- champs : Statistiques de l'édition
 - et, par verbete :
 - Mot
 - ClasseMot (éventuellement)
 - Sens
 - ClasseSens (éventuellement).

d) définition des champs :

Il s'agit de la première esquisse du dictionnaire de variables du système PC-DICO, construit à partir des données identifiées dans le projet logique. Pour chaque variable on indique la structure (afin d'identifier les champs composés), un intervalle de validation, un masque d'édition et l'origine, sous la forme d'une nomenclature de flux (à remplacer par un numéro d'écran ou de formulaire au cours de l'étape de spécification). Le tableau 1 présente le dictionnaire de variables construit pour cette version du logiciel.

e) algorithmes :

- A01 : Si Prefixe1 est non blanc, et Prefixe2 est blanc alors Prefixe2 = Prefixe1 ;
- A02 : Si Suffixe1 est non blanc, et Suffixe2 est blanc alors Suffixe2 = Suffixe1.

f) dépôt de données :

La structure logique de l'information stockée et manipulée par le système PC-DICO est représentée sous la forme d'un diagramme de relations entre entités (chapitre 2) :

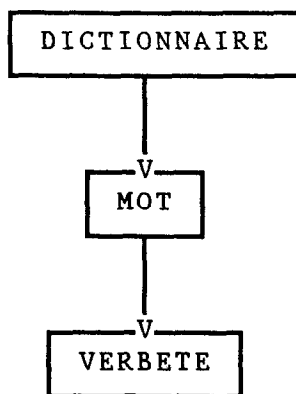
Tableau 1 : La phase de conception produit une première ébauche du dictionnaire de variables.

SYMBOLE	DESCRIPTION	Stru- cture	Inter- valle	Masque	Ori- gine	Cohé- rence
Mot	Mot de la lingua geral	1.01		X(36)	1.E	*
ClasseMot	Classe grammaticale du mot	2.01	T01	X	1.E	
Sens	Signification du mot (port.)	3.01	-	X(36)	1.E	
ClasseSens	Classe grammaticale du sens	4.01	T01	X	1.E	
Exemple	Texte, exemple d'util. du mot	5.01	-		1.E	
Ligne(Ex.)	Ligne du texte exemple	5.02		X(78)		
Etymolo.	Texte, etymologie du mot	6.01	-		1.E	
Ligne(Et.)	Ligne du texte etymologie	6.02		X(78)		
Critere	Critere de sélection d'édition	7.01			2.E	
Mot1	Début de sélection sur le mot	7.02	-	X(36)		
Mot2	Fin de sélection sur le mot	7.03	-	X(36)		
Prefixe1	Debut de sélection de prefixe	7.04	-	X(10)	A01	
Prefixe2	Fin de sélection de prefixe	7.05	-	X(10)	A01	
Suffixe1	Debut de sélection de suffixe	7.06	-	X(10)	A02	
Suffixe2	Fin de sélection de suffixe	7.07	-	X(10)	A02	
Classe	Classe grammat. sélectionnée	7.08	T01	X		
TailleMot	Taille choisie pour le mot	7.09	01 à 36	99		
FormeEtat	Forme de la sortie diction.	8.01	C.S	X	2.E	
Destinat.	Périphérique de sortie	9.01	E.I	X	2.E	
Ordre	Ordre de classement de sortie	10.01	L.P	X	2.E	

* dans le cas où le mot se présente avec des accents, le nombre de caractères utilisables est ramené à 36 moins le nombre d'accents ;

T01 désigne la table des classes grammaticales présentée ci-après (elle peut varier d'une langue à l'autre) :

code :	A	signification :	ADVERBIO
	C		CONJUNCAO
	D		SUBST.DEP.
	F		SUFXIO
	G		PART.GRAM.
	J		ADJETIVO
	N		NUMERAL
	P		PRONOME
	R		PREPOSICAO
	S		SUBST.
	V		VERBO



On appelle "*Dictionnaire*" l'ensemble des relations connues entre tous les vocables décrits de la *lingua geral* (appelés "*mot*") et les significations correspondantes en portugais, appelées "*sens*". On convient de désigner sous le nom de "*verbete*" le registre d'un sens pour un mot déterminé : le dictionnaire peut alors être considéré, indifféremment, comme une collection de verbetes, ou une collection de mots (regroupant toutes les verbetes qui ne sont que plusieurs sens attribués au même mot).

3.4. Commentaires

Pour compléter le projet logique du logiciel PCIDICO, il suffit d'établir une tactique de réalisation (chapitre 5) et de faire une estimation de la configuration minimale pour opérer le système (chapitre 6).

Bien que particulièrement simple, cet exemple appelle quelques remarques :

- on remarquera que le projet logique est volontairement maintenu à un niveau conceptuel très général, en évitant de représenter des détails susceptibles de figer certains choix physiques. Par exemple, on pourra constater que l'on n'a pas précisé de quelle façon la table des classes grammaticales doit être manipulée (une gestion complète pouvant faire l'objet d'une fonction importante), ou encore on n'a pris aucune décision sur la sécurité des fichiers (fonction de copie/récupération), qui pourra donc être intégrée, ou non, au progiciel développé ;

- lorsque l'on définit deux entités externes ayant des prérogatives différentes, comme c'est le cas entre le linguiste et l'utilisateur, on sous-entend qu'un contrôle d'accès devra être implémenté, sous une forme ou une autre ;

- la description de la logique des fonctions n'est faite que lorsqu'elle est réellement nécessaire. Il eût été peu rationnel, par exemple, de décrire le détail du processus de mise à jour du fi-

chier dictionnaire, même à niveau logique, dans la mesure où le responsable de l'implémentation sait parfaitement de quoi il s'agit ! Il faut néanmoins tempérer cette observation, dans la mesure où cette logique n'apparaît qu'à cet endroit, l'utilisateur pouvant y trouver une utilité.

Ce logiciel a été programmé durant le mois de mai 1987 et une première version implantée début juin. Quelques modifications de détail ont ensuite été réalisées, de sorte que c'est une version 1.3 qui a finalement été exploitée en vraie grandeur, à partir du mois d'août pour préparer la publication du dictionnaire de "*lingua general*".

Le développement d'une nouvelle version, incluant en particulier la définition et la gestion complète d'un alphabet, sur l'écran comme sur l'imprimante, est envisagée dans un futur proche.

CONCLUSION

La méthode qui a été décrite est utilisée au sein du Département d'Informatique de l'EMBRAPA, depuis la fin 1984. Initialement, elle a été employée pour la réalisation de systèmes administratifs : comptabilité, patrimoine, etc..

Son application à des systèmes de soutien à la recherche, et donc de caractère scientifique, est une innovation due à l'équipe EMBRAPA/ORSTOM du programme SISGEO. Après les quatre premiers projets logiques (climat, végétation, documents cartographiques et images de satellites) déjà conclus, trois autres sont en cours de réalisation avec la même philosophie : sols, données socio-économiques et données phytosanitaires.

Parallèlement, la même méthodologie est utilisée dans le domaine génétique par d'autres équipes, pour des applications sensiblement différentes. L'expérience purement ORSTOM, acquise avec PCDICO, est en cours de renouvellement, avec un système très particulier, et plus complexe, destiné à rendre opérationnelle sur PC-compatible une méthode de synthèse et critique de données, mise au point par un chercheur de l'Institut.

La généralisation progressive de cette méthode, ou de toute autre proposition susceptible d'aboutir à la réalisation de modèles logiques pour les progiciels spécifiquement développés, pourrait constituer un guide précieux pour ce genre d'activités, pour l'instant plutôt artisanales. Dans cette optique, un cours destiné aux développeurs pourrait être préparé, et un guide rédigé et proposé comme recommandation de documentation de systèmes.

D'ores et déjà, il existe plusieurs produit-logiciels disponibles pour assister les professionnels dans la réalisation des travaux afférents à cette phase du développement d'un progiciel : ceux-ci donnent une nouvelle dimension à cette étape, principalement en permettant l'intégration de ses résultats pour la spécification et la codification. On peut raisonnablement en attendre des gains spectaculaires de productivité et qualité.

REFERENCES BIBLIOGRAPHIQUES

AFNOR : *Recommandation de Plan Qualité Logiciel*. Fascicule de documentation de la Normalisation Française, Z 67-130, 1987.

BOURGEOIS, Jacques : Ateliers de génie logiciel : état de l'art et perspectives. In : *Revue du Génie Logiciel*, 1, 1984. Agence de l'Informatique, Paris.

DE MARCO, Tom : *Structured Analysis and System Specification*. Yourdon Inc., New York, 1978.

EMBRAPA : *Manual de normas e procedimentos para o desenvolvimento de sistemas de informação*. EMBRAPA/DIN, Brasília, 1985. Diffusion restreinte.

GANE, Chris : *Rapid System Development using structured analysis and relational technology*. IBPI Instituto Brasileiro de Pesquisa em Informatica, Rio de Janeiro, 1987.

GANE, Chris & SARSON, Trish : *Structured Systems Analysis : tools and techniques*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1979. 256p.

MARTIN, James : Desenvolvimento de aplicações sem programadores. In : *Cadernos de informatica*, 4, 1981, Compucenter, Sao Paulo.

WARNIER, Jean-Dominique : *LCS Logica de construção de sistemas*. Datamec/Editora Campus Ltda, Rio de Janeiro, 1985. 191p.

YOURDON, Edward & CONSTANTINE, Larry : *Structured Design*. Yourdon Press, New York, 1978.

Session 3 :
BASES DE DONNEES DOCU-
MENTAIRES

**POPCAM, BASE DE DONNÉES BIBLIOGRAPHIQUES
INFORMATISÉE
SUR LES ÉTUDES DE POPULATION AU CAMEROUN :
ÉVOLUTION ET PERSPECTIVES**

Patrick Gubry
(Démographie)

CEPED - Centre Français sur la Population et le Développement
15, rue de l'École de Médecine
75006 Paris

RESUME - L'auteur a mis sur pieds une base de données bibliographiques sur les études de population au sens large, incluant - outre la démographie *stricto sensu* - un grand nombre de références se réclamant d'autres disciplines des sciences sociales, médicales et juridiques. Cette base compte en juin 1987 près de 2 700 références, toutes sur le Cameroun. Elle fonctionne - ou fonctionnera prochainement - dans six implantations au Cameroun et deux en France. L'auteur souhaite confronter cette expérience avec des expériences similaires, tant pour optimiser le système mis en place, que pour envisager son évolution.

1. PRESENTATION GENERALE DU SYSTEME

POPCAM est une base de données bibliographiques informatisée sur les études de population au Cameroun. Elle est actuellement en cours d'implantation sur des micro-ordinateurs compatibles IBM-PC avec le logiciel Texto dans six organismes situés au Cameroun et en France. Cette base, consacrée exclusivement au Cameroun, a vocation pour regrouper toutes les références bibliographiques du domaine de la démographie.

En outre, un certain nombre de disciplines complémentaires à la démographie figurent dans POPCAM, telles que l'histoire, la géographie, l'anthropologie, la sociologie, la linguistique, l'économie, le droit, la psychologie et la médecine. Les références concernant ces dernières disciplines sont sélectives et cela d'autant plus que l'on s'écarte du thème central de la base.

POPCAM comporte plus de 2 750 références en octobre 1987. Elles sont partiellement commentées et réparties en onze chapitres et 99 sous-chapitres ou thèmes (Cf. annexe 1).

2. OBJECTIFS D'UNE BASE DE DONNEES BIBLIOGRAPHIQUES SPECIALISEES

L'idée d'une base de données bibliographiques spécialisée s'est progressivement imposée à travers l'observation des conditions objectives de travail du chercheur dans les structures de recherche. Une solide base bibliographique est avant tout un outil indispensable à cette recherche, que ce soit pour le chercheur lui-même ou pour les structures initiant et abritant les opérations de recherche, et cela à cinq niveaux : l'élaboration de la politique scientifique, la définition des opérations de recherche ; l'analyse des résultats, les publications scientifiques et la valorisation des résultats.

La politique scientifique d'un pays donné doit pouvoir tenir compte à la fois des besoins et des priorités exprimés par les pouvoirs publics et les partenaires sociaux, des lacunes clairement identifiées au niveau national dans les connaissances relatives à un domaine donné, des courants scientifiques internationaux et des possibilités de mise en oeuvre et de financement. Une base de données bibliographiques est à même de répondre à la seconde de ces exigences, en permettant une recherche bibliométrique sur l'ensemble des travaux passés, afin de mettre en lumière les acquis et les faiblesses des résultats disponibles et de proposer des orientations futures (Gubry, P. & Lemdja, F., 1985a). Elle peut influencer par là directement les instances chargées de l'élaboration de la politique scientifique, aussi bien que l'ensemble des pouvoirs publics dans la définition de leurs priorités.

Au niveau des structures de recherche, la définition des opérations de recherche doit (devrait ?) obligatoirement tenir compte de l'ensemble des acquis du domaine considéré. Il s'agit à la fois d'éviter les redondances et de faire progresser les connaissances par l'intégration des résultats déjà obtenus auparavant. Cela demande un important travail préalable de recherche bibliographique, s'étendant sur plusieurs mois et dans plusieurs pays. Il est clair que le manque de moyens fait que ce travail tourne généralement court. La part qui en est effectuée entraîne quant à elle un gaspillage de ces moyens déjà limités, dans la mesure où ce travail est souvent recommencé d'une opération à l'autre, sans que l'on puisse toujours bénéficier des efforts déjà accomplis. Une base de données bibliographiques actualisée, entourée d'un minimum de publicité, est susceptible de répondre à ces besoins de manière quasi-instantanée.

L'analyse des résultats d'une enquête effectuée dans le cadre d'une opération de recherche en sciences sociales demande la prise en compte des acquis pour "dégager des éléments explicatifs d'un

phénomène observé, à partir des résultats obtenus dans d'autres travaux ou dans des disciplines voisines" (Gubry, P. & Lemdja, F., 1985b). L'observation du passé permet en outre de formuler des hypothèses quant à l'évolution prévisible. Ce problème se pose avec plus d'acuité encore dans les pays qui disposent d'un nombre restreint de travaux, où chacun de ceux-ci doit donc être valorisé au maximum.

En prolongement, toute publication scientifique comporte logiquement une bibliographie spécialisée plus ou moins étoffée. L'élaboration pratique de celle-ci est un investissement non négligeable : vérification et correction des références, complétude, mise en forme, dactylographie, etc., toutes opérations qu'une base de données a déjà réalisées par avance, ou qu'elle peut effectuer dans les délais les plus courts.

Enfin, force est de relever le nombre considérable de travaux scientifiques, pourtant dûment financés, qui tombent rapidement dans l'oubli. La raison en est certes dans le faible intérêt de certains d'entre eux (cependant jamais nul), mais aussi dans l'absence ou l'insuffisance des structures et des dispositifs de valorisation. Il en résulte un gaspillage de ressources absolument hors de mise. Une base de données bibliographiques a donc un rôle éminent à jouer dans la diffusion et la valorisation des résultats de la recherche, ce qui est un point crucial aussi bien pour le chercheur que pour l'organisme de recherche, car il légitime leur existence et conditionne par là leur fonctionnement.

Dans un domaine proche de la recherche, une telle base rend des services pour l'enseignement, à la fois aux enseignants pour la mise à jour de leurs connaissances et la préparation de leurs cours, et aux étudiants pour la préparation de leurs thèses.

Mais l'intérêt d'une base de données bibliographiques dépasse de loin les domaines de la recherche et de l'enseignement. Les divers secteurs de la planification et les ministères techniques s'occupant de population sont directement concernés et sont amenés à utiliser la base tant pour les besoins d'études ponctuelles, que pour la planification à long terme. On peut citer à cet égard le Plan (notamment, en l'occurrence, l'Unité de Planification de la Population), la Santé, l'Education Nationale, les Affaires Sociales, la Condition Féminine, l'Urbanisme, etc..

Un certain nombre d'utilisateurs du secteur privé sont par ailleurs susceptibles de faire appel à la base. Il s'agit des bureaux d'études bien sûr, auxquels on peut assimiler la plupart des missions des organismes internationaux, mais aussi un certain nombre d'entreprises privées qui utilisent des données démographiques ou socio-démographiques pour leurs études de marché. La plupart de ces utilisateurs recherchent des données numériques brutes, mais

la base leur fournit dans ce cas les "adresses" où ces données peuvent se trouver.

Enfin, il est question depuis un certain temps de mettre sur pieds une bibliographie nationale du Cameroun, dans le cadre d'une politique de recherche documentaire, d'acquisition d'ouvrages et de "mise en valeur du patrimoine national". Toute base bibliographique spécialisée peut donc être d'un grand secours dans ce domaine.

3. HISTORIQUE DU SYSTEME

POPCAM trouve son origine dans un fichier manuel individuel de chercheur établi progressivement entre 1972 et 1979. Ce fichier a été transcrit sur des fiches à perforations marginales au format standard 12,5 x 7,5 cm., approvisionné de 1980 à 1983. Les perforations n'ont cependant jamais été mises en oeuvre devant la montée de la micro-informatique.

A partir de 1983, différents facteurs se sont conjugués au sein des structures camerounaises de recherche, au Centre de Recherches Economiques et Démographiques (CRED) : acquisition d'un micro-ordinateur IBM 23 ; mise à disposition d'un ingénieur en informatique ; possibilités de financements complémentaires via le "Futures Group", un organisme américain, dans le cadre de la mise en place d'une Unité de Planification de la Population (UPP) au Ministère du Plan à Yaoundé.

Cette coopération a donné deux résultats concrets : l'informatisation de POPCAM au sein du CRED sur micro-ordinateur IBM 23, avec un "logiciel-maison" appelé PROCAM (Gubry, P. & Lemdja, F., 1985b) et la publication d'un ouvrage bibliographique, correspondant au contenu - développé - de la base, avec 2 472 références en 1984 (Gubry, P. 1984).

Cet aboutissement a été conditionné par la constatation que POPCAM apporte quelque chose de fondamentalement nouveau dans son domaine par rapport à ce qui existe déjà par ailleurs, à savoir les ouvrages bibliographiques sur le Cameroun et les bases de données spécialisées extérieures, en l'occurrence américaines. Les premiers sont trop généraux, ce qui implique notamment la mise à l'écart d'une grande partie de la "littérature grise", ou informelle, et difficiles à actualiser. Les secondes sont beaucoup trop incomplètes pour répondre aux besoins nationaux (moins de 200 références sur le Cameroun à cette époque). POPCAM constitue ainsi la première base de données bibliographiques informatisée fonctionnant au Cameroun.

Malheureusement, l'implantation réalisée est devenue très rapidement caduque devant la standardisation croissante en micro-

informatique, autour du standard IBM-PC, au niveau mondial certes, mais encore plus au Cameroun. Par ailleurs, le logiciel PROCAM, conçu dans les délais les plus courts pour les besoins de la cause, compte tenu des moyens disponibles, avait des possibilités limitées.

Dans ces conditions, il a paru indispensable de transférer le système sur micro-ordinateur IBM-PC ou compatible, avec un logiciel du marché. Un transfert automatique des données n'a finalement pas été possible pour des raisons de compatibilité, tant au niveau du transfert lui-même, qu'au niveau du passage sur logiciel courant (dBASE). Plus précisément, un tel transfert aurait occasionné, pour un résultat égal, un "coût" de loin supérieur à une nouvelle saisie complète de la base. En effet, une nouvelle saisie permettait d'optimiser dès le départ la constitution et le fonctionnement du système en fonction de l'expérience acquise et des possibilités accrues offertes par le logiciel. C'est donc cette solution qui a été retenue.

Dans le choix du logiciel, plusieurs solutions ont été examinées, chacune présentant des avantages et des inconvénients. L'exigence impérative était pour le logiciel de posséder une version micro compatible avec IBM-PC, afin de permettre une diffusion aussi large et décentralisée que possible. Dans ce cadre, on a pris en compte les performances intrinsèques, la "*communicabilité*", la simplicité d'utilisation et le prix. C'est finalement le logiciel Texto qui a été retenu, à la fois pour les performances spécifiques qu'il offre dans le domaine de la documentation et pour assurer la compatibilité avec les systèmes en place ou en cours d'implantation (MESRES au Cameroun ; MINCOOP avec la base IBISCUS, ORSTOM avec la base HORIZON et CIRAD en France).

POPCAM est ainsi devenu opérationnel sous sa nouvelle forme (compatibilité IBM-PC et logiciel Texto) en avril 1987, avec l'aide de l'ORSTOM et du Groupe de Démographie du Développement (GDD).

4. DESCRIPTION DE POPCAM

La base POPCAM comporte 31 champs, qui sont détaillés en annexe. Un effort particulier a été fait pour homogénéiser la saisie, celle-ci étant appelée à se faire à l'avenir dans des organismes différents (Cf. instructions en annexe 2). Cela veut dire que les divers cas possibles ont été analysés au maximum. Il n'est toutefois pas exclu que des précisions supplémentaires soient ajoutées ultérieurement en tant que de besoin.

L'attention doit être attirée sur deux champs spécifiques : le champ NOUVREF et le champ SITU.

Le champ NOUVREF est destiné à répertorier des republications de la même référence sous d'autres formes (par exemple, un même article publié dans deux revues différentes). Dans ce cas, les descripteurs (thématiques, géographiques, etc.) restent exactement les mêmes et il n'y a pas lieu de créer un nouveau "document" dans la base, comme il serait de règle dans une bibliothèque possédant effectivement les différents ouvrages référencés. POPCAM est en effet *une base de recherche documentaire*, destinée à fonctionner dans divers organismes ne possédant pas nécessairement eux-mêmes les ouvrages originaux répertoriés dans la base. POPCAM cherche à couvrir le plus exhaustivement possible un domaine scientifique donné et répond par là essentiellement au premier des objectifs d'une recherche documentaire, qui est *l'identification* des ouvrages. Il s'agit là de l'originalité de POPCAM par rapport à la plupart des autres bases de ce type, qui répertorient généralement de manière exclusive les ouvrages en possession de l'organisme qui les met en oeuvre. Cela facilite l'accessibilité des ouvrages, mais s'exerce au détriment de l'exhaustivité de la base.

Avec POPCAM, l'accès effectif aux ouvrages se fait dans une étape ultérieure par l'intermédiaire du champ SITU, qui a pour vocation de donner la liste des centres de documentation possédant l'ouvrage en question, situés tant au Cameroun qu'à l'extérieur, quand ceux-ci sont connus.

Un effort important a été porté à la qualité et à l'homogénéité des données contenues dans POPCAM. C'est ainsi que la saisie initiale a donné lieu à six types de contrôles différents, soit à un nombre de contrôles pour chaque champ allant de trois à cinq :

- correction à la saisie avec la procédure "d'ajout avec vérification". Chaque "document" est relu, puis corrigé s'il y a lieu, avant son entrée dans la base ;

- première correction sur listing. Correction concomitante (toutes les 200 références) ;

- contrôle d'exhaustivité. Présence exhaustive des "documents" et complétude des sept champs qui doivent être obligatoirement remplis (ANNEE, LAN, CHAP, SCHAP, DES, PROV, DATE) ;

- contrôle sur index des dix-neuf champs suivants, regroupés pour la circonstance en dix-sept index : AU, INAU, AUC, INAUC, ANNEE, REVUE, LED, ED, LAN, CHAP, SCHAP, DES, CDES, PROV, LIEUX, ETH, CETH, DATE, SITU ;

- contrôle sur liste des quatre champs suivants : CONG, DIPL, COLL, NOUVREF. Ces champs peuvent contenir la virgule, séparateur "d'article", et ne sauraient donc être valablement "inversés" (associés à un index) ;

- deuxième correction sur listing. Correction finale (par groupe de 500 références).

Ces contrôles sont à recommencer régulièrement pour les nouvelles références introduites.

POPCAM fonctionne actuellement avec dix-neuf champs "*inversés*" dans seize index différents. La version micro du logiciel Texto (version 3.8 de juillet 1986) n'autorise pas, pour des raisons de capacité, que ces dix-neuf champs puissent être tous interrogés en "*mode inverse*", c'est à dire soient automatiquement interrogés via les index, plutôt qu'en "*mode séquentiel*", beaucoup plus long. Cela s'est cependant avéré impossible pour douze d'entre eux, parmi lesquels onze sont mis à jour automatiquement après ajout ou correction (Cf. tableau ci-après). En outre, l'index IDÈS regroupant les descripteurs et les candidats-descripteurs, qui est le plus utilisé, a été "*implicite*", ce qui veut dire que le nom du champ n'a pas besoin d'être rappelé lors de la formulation des interrogations. Bien entendu, si le besoin d'un autre champ à interroger en "*mode inverse*" se fait sentir, un autre "*document de paramètre*" Texto, prenant en compte le champ en question, peut être activé à tout moment.

5. IMPLANTATIONS ET MISE A JOUR

POPCAM est actuellement implanté dans trois sites différents : CRED à Yaoundé, ORSTOM à Bondy et Centre sur la Population et le développement (CEPED), en cours de constitution, à Paris. Quatre autres structures sont immédiatement intéressées : l'Unité de Planification de la Population (UPP) au Ministère du Plan à Yaoundé, l'Institut de Formation et de Recherche démographiques (IFORD), la Sous-Direction des Statistiques Démographiques et l'OCEAC. L'implantation du système doit enfin être envisagée au CEDID à Paris et dans divers organismes à Yaoundé : le Ministère de l'Enseignement Supérieur et de la Recherche Scientifique, l'Institut des Sciences Humaines, l'Université, le Ministère de la Condition Féminine, la Mission ORSTOM, etc.. Le principal obstacle à une telle diffusion est le coût du logiciel, alors que des logiciels à fonctions analogues sont diffusés gratuitement. La maintenance, voire l'acquisition du matériel est le second problème à résoudre. L'ambitieux projet d'informatisation de la documentation dans les instituts de recherche au Cameroun va contribuer à la solution de ces problèmes à partir de 1988.

Le système a été testé ou fonctionne actuellement sur des matériels très variés : IBM-XT, CORDATA-PPC400, IEEE-XTX Turbo, Thomson Microméga, Bull Micral 40, etc., avec des disques durs de 10 à 70 MO.

POPCAM a vocation à être approvisionné dans chacune de ses implantations. L'esprit qui a présidé à sa conception répond assez bien - mais pas totalement - aux besoins de la plupart des organismes cités. On ne saurait donc exclure que chaque utilisateur l'adapte encore plus à ses besoins spécifiques dans un but essentiellement pratique. Cela peut même être considéré comme souhaitable. Dans ces conditions, les mises à jour réciproques interviendraient régulièrement à travers un "réseau POPCAM", constitué par les différents utilisateurs, sans pour autant que chacun d'eux soit obligé de conserver la totalité des références saisies par les autres. Le champ PR est alors destiné à repérer l'origine d'un "document" figurant dans la base.

Il est clair qu'un effort constant de mise à jour doit être fourni si l'on veut conserver au système une fiabilité élevée. Aucune source d'information ne doit être négligée :

- les publications directes, dont les travaux des centres de recherche, au Cameroun et à l'extérieur ;
- les revues bibliographiques spécialisées ;
- les ouvrages bibliographiques, dont les répertoires de travaux et de thèses ;
- les travaux universitaires réalisés chaque année au Cameroun dans les disciplines contenues dans la base ;
- les centres de documentation (élargissement de la prospection et mise à jour) ;
- les bases de données bibliographiques extérieures ;
- les petits séminaires spécialisés, dont les communications restent souvent du domaine de la "littérature informelle" ;
- les autres services producteurs clairement identifiés.

Enfin, les contacts personnels directs ne doivent pas être sous-estimés. En particulier, un contact direct avec la cinquantaine de démographes travaillant actuellement sur le Cameroun peut permettre d'approvisionner continuellement la base de manière très directe et donc de faire connaître les travaux de ces derniers.

6. PERSPECTIVES

Au niveau de la base elle-même, l'objectif est dans l'immédiat de rendre le système plus "convivial" en faisant apparaître une série de menus, tant pour les ajouts, que pour la correction et l'interrogation. Deux possibilités existent à cet effet dans Texto : les "documents de dialogue" et le module Logotel. Les premiers ont déjà été testés.

Cependant, même en l'absence de ces écrans d'aide, un simple complément aux instructions peut permettre à quiconque de mettre en oeuvre instantanément les fonctions essentielles que sont les divers types de corrections, la saisie de nouvelles références et les interrogations simples. Le manuel d'instructions doit donc être complété à la fois par un *"mode d'emploi"* plus détaillé et par les listes, régulièrement mises à jour, des *"articles"* déjà entrés dans les champs principaux à caractère *"ouvert"* (AU/INAU, AUC/INAUC, CDES, LIEUX, ETH), de manière à conserver l'homogénéité.

POPCAM peut donner lieu à trois types de publications à l'avenir : des ouvrages bibliographiques actualisés, reprenant tout ou partie de la base, à des intervalles plus ou moins longs ; une *"note d'information"* pour échanger les expériences entre utilisateurs ; un *"bulletin bibliographique"* pour diffuser les nouvelles références.

Le système pourrait être perfectionné en recourant à la télématique (vidéotex interactif). La mise en route du réseau Campac au Cameroun, relié notamment à Transpac, rend la chose envisageable même au niveau international. L'équipement actuel des diverses structures intéressées ne permet cependant pas sa réalisation immédiate.

Sur un plan général, ce type de base pourrait être étendu à d'autres pays avec les mêmes disciplines scientifiques. Sa multiplication permettrait alors d'effectuer des recherches thématiques transversales, plus approfondies qu'avec les bases transnationales existantes.

Enfin, d'autres disciplines pourraient être intéressées par cette approche. Au Cameroun, certaines d'entre elles, possédant déjà un fond important, ne nécessiteraient qu'un investissement marginal faible pour leur informatisation, telle la linguistique ou la géographie.

Au niveau des organismes de recherche, comme l'ORSTOM, de très nombreux chercheurs possèdent leur propre fichier bibliographique spécialisé, qui offre des services très pratiques. L'informatisation de ces fichiers et leur mise à disposition de l'extérieur permettrait de rentabiliser les efforts déployés et de valoriser l'ensemble de l'organisme, notamment les services de documentation appelés à offrir des postes de consultation.

BIBLIOGRAPHIE

GUBRY (Patrick) - 1984 - *Bibliographie générale des études de population au Cameroun* (arrêtée au 31 mars 1984). General Bibliography of Population Studies in Cameroon (As at 31st March 1984). Yaoundé : Ministère de l'Enseignement Supérieur et de la Recherche Scientifique, 382 p..

(Travaux et Documents de l'Institut des Sciences Humaines ; collection Etudes Bibliographiques et Recherches en Bibliothéconomie, n° 8).

GUBRY (Patrick), LEMDJA (Francine) - 1985 a - Cent ans d'études de population au Cameroun (1884-1984). In : *Utilisation des statistiques démographiques au Cameroun*. Actes d'un séminaire tenu du 16 au 19 juillet 1984 à Yaoundé. Yaoundé : Ministère du Plan et de l'Aménagement du territoire, 381 p., pp. 57-82.

GUBRY (Patrick), LEMDJA (Francine) - 1985 b - POPCAM, base de données bibliographiques sur les études de population au Cameroun. Communication au Congrès général de l'Union Internationale pour l'Etude Scientifique de la Population, Florence (5-12 juin 1985). *Revue Science et Technique, série Sciences Humaines/ Science and Technology Review, Social Sciences Series* (Yaoundé), vol. III, n° 3-4, pp. 93-99.

POPCAM : Répartition des index par champs, possibilité d'interrogation et diverses caractéristiques.

POPCAM : Répartition des index par champs, possibilité d'interrogation et diverses caractéristiques.

index	champ(s)	interrogation en mode inversé	mise à jour différée automatique	longueur maximum des articles	nombre d'articles introduits (Septembre 1987)	durée appro- ximative de réalisation (À 4.77 MHz)
IAU	AU, INAU	+	+	35	1911	60 mn
IAUC	AUC, INAUC	+	+	46	214	90 mn
IANNEE	ANNEE	+	+	4	112	12 mn
IREVUE	REVUE	-	-	46	282	35 mn
ILED	LED	-	-	20	159	20 mn
IED	ED	-	-	46	363	33 mn
ILAN	LAN	-	-	2	7	10 mn
ICHAP	CHAP	-	-	24	11	12 mn
ISCHAP	SCHAP	-	-	36	99	30 mn
IDES	DES, CDES	+	+	36	295	70 mn
IPROV	PROV	+	+	2	11	10 mn
ILIEUX	LIEUX	+	+	30	403	25 mn
IETH	ETH	+	+	12	94	16 mn
ICETH	CETH	+	-	12	19	10 mn
IDATE	DATE	+	+	6	293	14 mn
ISITU	SITU	-	-	8	10	12 mn

N.B. : la longueur des articles prises en compte par un index est au maximum de 47 caractères. La longueur figurant ici est celle des index d'interrogation. Pour les corrections, une longueur plus grande a été utilisée pour les champs de longueur fixe. Pour les index qui ne sont pas en correction automatique, le nombre d'articles introduits correspond à ceux qui figurent dans les derniers index réalisés et non pas dans la base elle-même à la date indiquée.

**ANNEXE 1: REPARTITION DES REFERENCES PAR DESCRIPTEUR SELON
QU'IL S'AGIT DU THEME PRINCIPAL OU D'UN THEME SECONDAIRE
(Septembre 1987)**

CHAP	SCHAP et DES	THEME		TOTAL
		PRINCIPAL	SECONDAIRE	
BIBLIOGRAPHIE	OUVRAGE BIBLIOGRAPHIQUE	68	8	76
	REVUE BIBLIOGRAPHIQUE	6	-	6
METHODOLOGIE	RECENSEMENT ADMINISTRATIF	3	2	5
	METHODOLOGIE RECENSEMENT	23	8	31
	ETAT CIVIL	18	5	23
	METHODOLOGIE ENQUETE	16	6	22
	METHODOLOGIE EPR	21	9	30
	METHODOLOGIE AUTRE	9	7	16
	ANTHROPONYMIE	11	-	11
STATISTIQUES-GEOGRAPHIE	STATISTIQUES	39	27	66
	DICTIONNAIRE DES VILLAGES	37	-	37
	REPertoire	5	2	7
	TOPONYMIE	6	1	7
	ATLAS-ENCYCLOPEDIE	7	1	8
	ATLAS REGIONAL	10	-	10
	STRUCTURES AGRAIRES	10	-	10
GENERALITES	ETUDE REGIONALE	46	4	50
	ORGANISATION INTERNATIONALE	16	-	16
	ETUDE REGIONALE	19	1	20
	ETUDE REGIONALE EN	77	13	90
	ETUDE REGIONALE NO	17	19	36
	ETUDE REGIONALE AD	19	7	26
	ETUDE REGIONALE ES	19	8	27
	ETUDE REGIONALE CE	46	4	50
	ETUDE REGIONALE SU	12	9	21
	ETUDE REGIONALE LT	22	9	31
	ETUDE REGIONALE OU	39	9	48
	ETUDE REGIONALE NW	17	1	18
	ETUDE REGIONALE SW	11	-	11
	PEUPLEMENT	10	4	14
	PEUPLEMENT EN	27	12	39
	PEUPLEMENT NO	13	5	18
	PEUPLEMENT AD	17	7	24
	PEUPLEMENT ES	3	2	5
	PEUPLEMENT CE	21	3	24
	PEUPLEMENT SU	5	9	14
	PEUPLEMENT LT	4	3	7
	PEUPLEMENT OU	19	3	22
	PEUPLEMENT NW	14	2	16
	PEUPLEMENT SW	2	2	4
	ETAT DE LA POPULATION	4	5	9

ANNEXE 1 (suite)

CHAP	SCHAP et DES	THEME	THEME	TOTAL
		PRINCIPAL	SECONDAIRE	
NUPTIALITE-FAMILLE	DROIT DU MARIAGE	15	3	18
	DROIT COUTUMIER DU MARIAGE	18	3	21
	NUPTIALITE	37	16	53
	NUPTIALITE SPECIFIQUE	44	11	55
	FAMILLE	27	28	55
NATALITE-FECONDITE	FACTEURS SOCIAUX DE LA FECONDITE	40	23	63
	FACTEURS BIOLOGIQUES DE LA FECONDITE	32	10	42
	FECONDITE	25	14	39
	PLANIFICATION FAMILIALE	13	4	17
MORBIDITE-MORTALITE	STATISTIQUES SANITAIRES	13	-	13
	SANTE	51	11	62
	MEDECINE TRADITIONNELLE	23	4	27
	NUTRITION	11	29	40
	MALADIE INFECTIEUSE	20	15	35
	MALADIE PARASITAIRE	88	5	93
	MORBIDITE	35	15	50
	MORTALITE	18	20	38
	MORTALITE INFANTILE	84	24	108
MIGRATION-URBANISATION	MIGRATION INTERNE	83	31	114
	EXODE RURAL	52	37	89
	URBANISATION	33	40	73
	URBANISATION DLA	77	25	102
	URBANISATION YDE	74	35	109
	URBANISATION AUTRE	96	24	120
	MIGRATION EXTERNE	36	5	41
	DEPLACEMENT TEMPORAIRE	17	2	19
	STATISTIQUES DU TOURISME	5	-	5
PLANIFICATION-POLITIQUES	PLANIFICATION	28	21	49
	POLITIQUES DE POPULATION	31	62	93
	DROIT DU TRAVAIL	30	10	40
	PROTECTION SOCIALE	30	5	35
	FORMATION	10	2	12
	RECHERCHE	23	11	34

ANNEXE 1 (suite)

CHAP	SCHAP et DES	THEME	THEME	TOTAL
		PRINCIPAL	SECONDAIRE	
SOCIO-ECONOMIE	DEVELOPPEMENT	62	69	131
	ECONOMIE-POPULATION	23	7	30
	ECOLOGIE HUMAINE	6	6	12
	STATISTIQUES SCOLAIRES	34	-	34
	SYSTEME SCOLAIRE	57	4	61
	SCOLARISATION	30	7	37
	RENDEMENT SCOLAIRE	22	4	26
	EDUCATION FEMININE	6	4	10
	FORMATION SPECIALE	9	5	14
	FORMATION-EMPLOI	21	6	27
	STATISTIQUES DU TRAVAIL	16	-	16
	POPULATION ACTIVE	54	15	69
	EMPLOI SECTORIEL	37	10	47
	POPULATION ACTIVE AGRICOLE	28	34	62
	AGRO-INDUSTRIE	45	16	61
	RECENSEMENT INDUSTRIEL	10	-	10
	SECTEUR INFORMEL	32	9	41
	CONDITIONS DE TRAVAIL	24	6	30
	POPULATION ACTIVE FEMININE	30	19	49
	CONDITION FEMININE	72	46	118
DELINQUANCE	26	6	32	
SOUS-POPULATION	4	12	16	
CONJONCTURES-PERSPECTIVES	CONJONCTURE	35	38	73
	PERSPECTIVES	23	16	39
TOTAL		2730	1128	3858

Observation: il y a 11 chapitres et 99 sous-chapitres ou descripteurs.

**ANNEXE 2: INSTRUCTIONS POUR LA SAISIE ET
L'INTERROGATION
(POPCAM, version d'octobre 1987)**

La base de données bibliographiques informatisée POPCAM comporte 31 champs qui sont les suivants:

NUM	NUMERO DU DOCUMENT
AU	AUTEUR(S)
AUC	AUTEUR(S) COLLECTIF(S)
ANNEE	ANNEE DE PUBLICATION
TI	TITRE ORIGINAL
TIT	TITRE TRADUIT
INAU	EDITEUR(S) SCIENTIFIQUE(S) PERSONNE(S) PHYSIQUE(S) D'UN OUVRAGE COLLECTIF D'OU EST EXTRAIT LE TITRE
INAUC	EDITEUR(S) SCIENTIFIQUE(S) PERSONNE(S) MORALE(S) D'UN OU- VRAGE COLLECTIF D'OU EST EXTRAIT LE TITRE
INTIT	TITRE D'UN OUVRAGE COLLECTIF D'OU EST EXTRAIT LE TITRE
CONG	CONGRES
DIPL	DIPLOME
REVUE	TITRE DE LA REVUE
LED	LIEU(X) D'EDITION
ED	EDITEUR COMMERCIAL (EDITEURS COMMERCIAUX)
VOLNO	NUMERO DE VOLUME
PAGES	NOMBRE DE PAGES
COLL	COLLECTION
NOUVREF	NOUVELLE REFERENCE
LAN	LANGUE(S) DE L'OUVRAGE
CHAP	CHAPITRE
SCHAP	SOUS-CHAPITRE
DES	DESCRIPTEUR(S)
CDES	CANDIDAT(S) DESCRIPTEUR(S)
PROV	PROVINCE(S) DU CAMEROUN
LIEUX	AUTRES LIEUX GEOGRAPHIQUES
ETH	ETHNIE(S)
CETH	CANDIDAT(S) ETHNIE(S)
DATE	DATE DE CREATION DU DOCUMENT DANS LA BASE
PR	PRODUCTEUR(S) DU DOCUMENT DANS LA BASE
SITU	SITUATION PHYSIQUE DE L'OUVRAGE
CONTENU	CONTENU

Ne sont remplis que les champs concernés par la référence en question (= document Texto) et pour lesquels l'information est connue. Dans le cas contraire, le champ est laissé "vide".

Au moment de la saisie, les articles à saisir dans un champ donné doivent être confrontés à la liste des "articles" déjà introduits, si celle-ci existe, pour des raisons d'homogénéité totale de l'écriture et pour permettre l'interrogation ultérieure.

1. CHAMP NUM Numéro du document

Le "document", au sens du logiciel Texto, correspond à l'enregistrement, la fiche bibliographique ou encore la référence. Chaque document de POPCAM est numéroté automatiquement à l'aide d'un numéro de quatre chiffres par pas de un (exemple: 0101, 0102, 0103, etc.).

2. CHAMP AU Auteur(s)

Nom(s) et prénom(s) de l'auteur personne physique.

Le nom est écrit en majuscules. Tous les tirets (-) qui font éventuellement partie d'un nom, ou qui séparent deux noms, sont supprimés et remplacés par un espacement, à des fins d'homogénéité et pour faciliter la consultation.

Le prénom est écrit en écriture riche (majuscules et minuscules), à la suite du nom et entre parenthèses. Les tirets éventuels sont supprimés et remplacés par un espacement.

Si le prénom entier n'est pas connu, l'initiale (ou les initiales) est mentionnée seule, suivie d'un point et entre parenthèses.

Tous les titres, honorifiques ou autres, sont supprimés. Les particules sont reportées, s'il y a lieu, après le prénom à l'intérieur des parenthèses, en minuscules, ou placées en majuscules avant le nom, selon les règles internationales.

Pour maintenir l'homogénéité, et contrairement aux usages en cours dans telle ou telle partie du Cameroun, l'ordre suivant est adopté pour chaque document dans la transcription des noms et des prénoms: nom, nom du père (s'il y a lieu) (ou "surnom"), prénom(s). Les noms et prénoms retranscrits sont ceux figurant sur l'ouvrage. Ainsi deux noms différents peuvent parfois correspondre à la même personne, si celle-ci a volontairement modifié son nom. En revanche, si le même auteur a plusieurs ouvrages avec des renseignements moins complets concernant le nom ou le prénom sur l'un d'entre eux, ceux-ci sont complétés le plus possible et homogénéisés.

S'il y a plusieurs auteurs, les noms et prénoms de chacun d'eux sont séparés d'une virgule suivie d'un espacement.

Si l'auteur personne physique n'est pas connu, seul le champ AUC (*auteur collectif*) est rempli. Si l'auteur personne physique n'est connu qu'en dehors de toute mention dans l'ouvrage, on ne le signale que dans le champ *contenu*.

Le nom et le prénom sont suivis, s'il y a lieu, d'une ou des mentions suivantes :

- coll. collaborateur (avec la collaboration de, avec la participation de, avec l'assistance de) ;
- ed. éditeur scientifique ;
- dir. directeur scientifique (sous la direction de) ;
- trad. traducteur (traduit par) ;
- préf. préfacier (si celui-ci est indiqué en tête de l'ouvrage, sinon il peut être signalé dans le champ contenu.

Si certains auteurs ne sont pas connus, ou si les auteurs sont plus de dix, la mention "*et al.*" remplace leur nom, après le dernier auteur connu.

Exemple : SAINT VIL

MARONTATE (Janet), MAITRA (A.), LOOKY (Silvère), GIL (Benjamin) coll.

TABI ABODO (Alphonse)

LOGINOVA (Valentina P.)

COLOMBEL (Véronique de)

HAGEN (Günther T. von)

NKOUNGOUROU (Abel) dir.

LEBEUF (Jean Paul) préf.

NASAH (B.T.), et al.

3. AUC Auteur(s) collectif(s)

Nom de l'auteur collectif.

Il s'agit de l'auteur personne morale, d'un organisme. Le champ AUC n'est à remplir que si aucun auteur personne physique n'est porté sur la première page de l'ouvrage. Si l'auteur personne physique est mentionné dans le corps de l'ouvrage, les champs AU et AUC tous deux. Dans ce cas la mention ed. suit le nom de l'auteur (ou de chaque auteur) collectif.

Le nom est écrit en majuscules. Les virgules figurant éventuellement dans un intitulé sont supprimées et remplacées par un tiret accolé, car la virgule est un "*séparateur d'article*" à l'intérieur d'un "*champ*".

S'il y a plusieurs auteurs collectifs, les noms de chacun d'eux sont séparés d'une virgule suivie d'un espacement.

Les organismes administratifs des pays sont précédés du nom du pays libellé en français, suivi de deux points entre espacements. Il en est de même des organismes des Nations-Unies, précédés par "*Nations-Unies*", ou des organismes "*sous-régionaux*" (UDEAC). Exception est faite pour les organismes de recherche et les universités, mentionnés en tant que tels. Le sigle est alors privilégié et précède le nom, séparé de ce dernier par un point. Pour les structures peu connues au Cameroun, le nom du lieu est ajouté à la fin après un point.

Dans le cas des ouvrages anonymes (ne comportant donc pas d'auteur personne physique), dont l'auteur collectif AUC est difficile à identifier, on prend comme auteur collectif le titre de la revue dans le cas d'une revue et le nom de l'éditeur commercial pour un ouvrage. Ces noms sont suivis respectivement, après un point, de la mention REVUE ou de la mention EDITEUR.

Exemple : CAMEROUN

CAMEROUN : SOUS-DIRECTION DES ETUDES DE POPULATION
 UDEAC : BUREAU TECHNIQUE REGIONAL DU PROJET DEMOGRAPHIQUE
 UDEAC-TCHAD ed.
 IPD. INSTITUT PANAFRICAIN POUR LE DEVELOPPEMENT
 INSD. INSTITUT NATIONAL DE LA STATISTIQUE ET DE LA DEMOGRA-
 PHIE. OUAGADOUGOU
 AMTSBLATT FUR DAS SCHUTZGEBEIT KAMERUN. REVUE
 FOYER D'EDUCATION ET D'UNION. EDITEUR

4. ANNEE Année de parution

Date de publication.

Si la date précise n'est pas connue, la date approchée est suivie de ca (circa).

La date est suivie de "*depuis*" pour les titres de revues bibliographiques mentionnées en tant que telles dans la base, qui contiennent à paraître.

S'il s'agit d'une période, les deux dates extrêmes sont séparées par un tiret.

Exemple : 1974

1971ca

1975depuis

1971-1978

5. TI Titre originel

Le titre est retranscrit comme sur l'ouvrage.

Le champ TI peut contenir une traduction du titre après un /, si l'ouvrage lui-même porte également le titre traduit.

Quand plusieurs volumes constituent manifestement une unité, ils peuvent être référencés dans le même "*document*". Le champ TI contient alors le titre de chacun des volumes.

Le champ TI se termine par un point.

Exemple : Peoples of the Central Cameroons.

Dictionnaire des villages de la Manyu/Village Dictionary
 of Manyu Division

Banyang Settlements. Part I : Pre-European Settlements.

Part II : Change of Settlements and Latter developments.

L'état civil à Douala.

Die Bana.

6. TIT Titre traduit

Il s'agit de la traduction du titre en français, pour les langues autres que l'anglais. Ce champ n'est donc pas rempli si le titre est en français ou en anglais, les deux langues officielles du Cameroun.

Le champ TIT se termine par un point.

7. INAU Editeur(s) scientifique(s) personne(s) physique(s)

Ce champ est rempli si la référence correspondante au titre est un chapitre d'un ouvrage collectif. Les champs INAU, INAUC et INTI contiennent par conséquent les caractéristiques de l'ouvrage collectif (In = tiré de).

Nom et prénoms de l'éditeur scientifique personne physique de l'ouvrage collectif d'où est extrait le titre, suivi par ed. ou dir.

Lorsqu'il y a plusieurs éditeurs scientifiques personnes physiques, comprenant collaborateur(s), traducteur(s) ou préfacier(s), seuls sont mis les éditeurs principaux, suivis chacun par ed. Il n'y a donc qu'un seul qualificatif possible par personne, celui-ci étant ed. (éditeur scientifique).

Le champ INAU est écrit comme le champ AU.

8. INAUC Editeur(s) scientifique(s) personne(s) morale(s)

Nom de l'éditeur scientifique personne morale de l'ouvrage collectif d'où est extrait le titre, suivi par ed.

Le champ INAUC est écrit comme le champ AUC.

9. INTI Titre d'un ouvrage collectif d'où est extrait le titre

Ce champ est écrit comme le champ TI, mais n'est jamais traduit, comme cela se passe dans le champ TIT.

Il se termine par un point.

10. CONG Congrès

Il s'agit de toutes les réunions scientifiques : congrès, colloque, réunion, séminaire, symposium, table-ronde, journée de travail, consultation régionale, voyage d'étude, stage, atelier, assemblée, etc.. ce champ est rempli à la fois pour les simples communications à des colloques, non édités ultérieurement et pour les ouvrages publiant les actes des colloques.

Ce champ contient le titre du congrès, le nom des organismes organisateurs (entre parenthèses), le lieu du congrès, la date du congrès (entre parenthèses), la référence du document s'il y a lieu.

Si la même communication a été faite à plusieurs congrès, ceux-ci sont mentionnés avec pour séparateur l'abréviation "&".

Le champ CONG se termine par un point.

Exemple : Colloque sur l'Observation Permanente et l'état civil, Libreville (13-17 décembre 1974), Doc. 62/SG/BTR-PD.
 Quatrième colloque de démographie africaine, Ouagadougou (20-24 janvier 1975).
 Deuxième stage de perfectionnement pour démographes africains francophones (IFORD, MINCOOP, UIESP), Bordeaux (8 novembre - 3 décembre 1982).

11. DIPL Diplôme

Il s'agit de tous les diplômes universitaires : thèses, mémoires, études de cas, etc.. ce champ est rempli chaque fois que le titre correspond à un diplôme, qu'il s'agisse de la version de souenance ou de la publication imprimée ultérieure, si le diplôme y est mentionné. Si le diplôme n'y est pas mentionné, mais que l'information est connue par ailleurs, elle est indiquée dans le champ CONTENU.

Le champ DIPL contient le titre du diplôme, l'université correspondante et, entre parenthèses, la date du diplôme, si celle-ci diffère de l'année de publication.

Il se termine par un point.

Exemple : Mémoire de maîtrise de géographie, Université de Paris VI.
 Thèse de doctorat de 3ème cycle d'histoire, Université de Yaoundé.
 Mémoire de licence en droit, Université de Yaoundé.
 Mémoire de fin d'études, IFORD, Yaoundé.
 Thèse de doctorat d'état de géographie, Université de Bordeaux III (1980).

12. REVUE Titre de la revue

Concerne les articles de revue et les revues bibliographiques citées en tant que telles.

Le titre de la revue est retranscrit comme sur la revue. Les majuscules sont privilégiées en cas de doute. Les virgules figurant éventuellement à l'intérieur du titre de la revue sont supprimées pour éviter la création d'un "article" artificiel à l'intérieur du champ.

Pour les bulletins internes à des organismes, le titre de la revue est suivi par le nom ou le sigle de l'organisme, après un point sans espacement.

Si la revue comporte un sous-titre, celui-ci est développé après le titre de la revue, à la suite d'un point sans espacement.

Le champ REVUE peut contenir une traduction du titre de la revue après un /, si la revue elle-même comporte cette traduction.

Exemple : Revue Science et Technique sér. Sci. Hum.

Population

Bulletin de liaison des Sciences Humaines. ORSTOM

Africa. Journal of the International Institute for African Languages and Cultures

Revue Science et Technique/ Science and Technology Review.

13. LED Lieu(x) d'édition

Nom de la ville de l'édition dans la langue originelle du pays, ou en français si le nom figure en français sur l'ouvrage. Nom du pays de l'édition entre parenthèses, si la ville est inconnue.

Le nom de la ville est suivi du nom du pays entre parenthèses, si elle est peu connue et des initiales de l'état pour les Etats-Unis, lorsque la précision apparaît utile.

S'il y a plusieurs lieux d'édition, les noms sont séparés d'une virgule suivie d'un espacement.

Le champ LED n'est pas rempli pour les congrès et les diplômes (portés dans les champs CONG et DIPL), s'il n'y a pas eu d'édition ultérieure (cas des simples communications à des colloques et des exemplaires de soutenance des thèses). Le lieu est alors déjà mentionné dans les champs CONG et DIPL.

Exemple : Yaoundé

Meudon (France)

Princeton (N.J.)

London, Paris.

14. ED Editeur commercial (éditeurs commerciaux)

Nom originel de l'éditeur.

Le champ ED n'est pas rempli pour les congrès et les diplômes (portés dans les champs CONG et DIPL), s'il n'y a pas eu d'édition ultérieure. Il n'est pas rempli non plus pour les articles de revue (nom de revue porté dans le champ REVUE), ni si l'auteur collectif lui-même (porté dans le champ AUC ou INAUC) est l'éditeur commercial. Cependant, s'il y a plusieurs auteurs collectifs et si l'un seul d'entre eux est éditeur commercial, ce dernier est néanmoins indiqué pour éviter toute confusion.

Si l'éditeur commercial est un organisme universellement connu par son sigle, celui-ci figure seul. Si l'organisme est peu connu, le développé du sigle suit ce dernier, après un point sans espacement.

Toute virgule figurant éventuellement à l'intérieur de l'intitulé d'un éditeur commercial est remplacée par un tiret accolé au mot suivant, pour éviter la création d'un "article" artificiel à l'intérieur du champ.

Si l'auteur lui-même est l'éditeur commercial, la mention "*chez l'auteur*" (ou chez les auteurs) est portée.

S'il y a plusieurs éditeurs commerciaux, les noms de chacun d'eux sont séparés d'une virgule suivie d'un espacement.

Exemple : PUF

UDEAC

Oxford University Press

CIDEP. Centre d'Information, de Documentation et d'Etude du Plan.

15. VOLNO Numéro de volume

Il s'agit du numéro de volume et du numéro d'ordre à l'intérieur du volume. Ce champ concerne les revues et les ouvrages à plusieurs volumes.

Ce champ est écrit selon les mêmes caractères et selon les mêmes conventions que sur l'ouvrage original.

Dans le cas où plusieurs volumes sont référencés dans le même "*document*" dans la base, le titre de chaque volume figure dans le champ TI et le champ VOLNO n'est pas rempli si les informations correspondantes figurent aussi dans le champ TI. Dans le cas contraire, les références de chacun des volumes sont relevées dans le champ VOLNO, séparées d'un point-virgule.

Exemple : vol. 1, n 2

vol. 34, 3

n 5, tome I

tome 1, n 21-22

4 vol.

LXII, article n 175 ; LXII, article n 196.

16. PAGES Nombre de pages

Ce champ reproduit la pagination de l'ouvrage.

Pour un ouvrage se composant d'un volume, ce champ contient le nombre de pages du volume suivi par l'abréviation p. après un espacement.

S'il y a plusieurs volumes, les paginations de chacun d'eux (écrites comme ci-dessus) sont séparées par des virgules suivies d'un espacement. Dans ce cas, si l'on connaît la pagination de certains volumes seulement, la mention "x" remplace le nombre de pages pour les volumes où celui-ci est inconnu.

Le champ PAGES peut contenir une ou plusieurs des mentions suivantes : multig. (multigraphiées), dactyl. (dactylographiées), manus. (manuscrites), annexes, cartes, h.t. (hors-texte), pagination multiple, non paginé, etc. (voir les exemples de présentation).

En cas de pagination multiple, si chacune des paginations est connue, les paginations sont séparées par le signe + entre deux espacements. Les paginations inconnues sont remplacées par la mention "x". Si toutes les paginations sont inconnues, la mention "*pagination multiple*" figure seule. Si la pagination multiple concerne le même écrit (par exemple, lorsque la préface ou la postface sont numérotées à part), l'abréviation p. apparaît une seule fois à la fin. En revanche, s'il s'agit d'écrits différents, rassemblés dans le même volume, l'abréviation p. est répétée pour chacun d'eux. Dans tous les cas, la nature de la numérotation (chiffres arabes ou romains) est conservée.

Dans le cas d'un volume à pagination multiple, la mention multig., dactyl. ou manus. est placée une seule fois à la fin. En revanche, lorsqu'il y a plusieurs volumes, la mention correspondante est répétée pour chaque volume, s'il y a lieu.

Pour un chapitre issu d'un ouvrage collectif, le champ PAGES comprend le nombre de pages de l'ouvrage collectif, suivi de la pagination du chapitre, après une virgule suivie d'un espacement. cette pagination est constituée des numéros des pages extrêmes du chapitre, séparés par un tiret et placés après l'abréviation pp. suivie d'une espacement.

Dans un article de revue, le champ PAGES est constitué des numéros des pages extrêmes de l'article dans la revue, séparés par un tiret et placés après l'abréviation pp. suivie d'un espacement.

Dans le cas d'un article à une seule page, le numéro de la page suit l'abréviation p., suivie d'un espacement.

La champ PAGES se termine par un point.

Exemple : Cas d'un ouvrage à un volume à pagination simple

128 p.
103 p. manus.
57 p. multig.
26 p. dactyl.
25 p. + annexes.
36 p. + 1 carte h.t.
non paginé.

Cas d'un ouvrage à pagination multiple concernant le même écrit

V + 42 p.

Cas d'un ouvrage à pagination multiple concernant des écrits différents

31 p. + 31 p.
65 p. + 95 p. multig.

Cas d'un ouvrage à plusieurs volumes

50 p., 219 p., x p., 151 p.
51 p. multig., 155 p. multig., 57 p. multig.

Cas d'un chapitre issu d'un ouvrage collectif

447 p., pp. 203-224.

Cas d'un article de revue

pp. 13-43.

Cas de deux articles de revue référencés dans le même "document"

pp. 99-103, pp. 115-119.

Cas d'un article à une seule page

p. 43.

17. COLL Collection

Il s'agit de la collection dans laquelle l'ouvrage est paru chez l'éditeur commercial.

Ce champ est écrit entre parenthèses.

Il contient aussi le nom de la sous-collection, s'il y a lieu (après un point-virgule) et le numéro d'ordre dans la collection ou la sous-collection (après une virgule).

Il contient enfin les références codées des documents des organisations internationales (sauf si les renseignements correspondants figurent déjà dans le champ CONG) et les numéros des contrats de recherche, s'ils figurent sur la page de titre.

Le champ COLL se termine par un point.

Exemple : (Travaux et Documents de l'Institut des Sciences Humaines :

Collection Etudes Bibliographiques et Recherche en
Bibliothéconomie, n 8).

(Changing African Family Project Monograph Series, n 4).

(Collection Architectures traditionnelles).

(ILO-OIT/TAP/CAMEROUN/R.10).

(Mémoires Africaines).

(Contract n AID/OTR-5412-C-00-1205-00).

18. NOUVREF Nouvelle référence

Concerne une republication, sous une autre forme, d'une publication précédente. Le champ NOUVREF n'est pas rempli si la publication se présente sous une forme manifestement provisoire : l'ancienne référence est alors éliminée et remplacée par la nouvelle.

Qu'il s'agisse d'une réédition pure et simple, ou d'une nouvelle publication du même texte chez un autre éditeur, le champ NOUVREF ne contient que les éléments qui ont changé par rapport à l'édition antérieure. cela peut cependant conduire à une nouvelle rédaction complète de la référence, qui est alors présentée sous une forme classique (voir exemples).

Ce champ peut donc contenir en plus le nom de l'auteur personne physique, si celui-ci ne figure pas dans la référence antérieure. Ce nom est placé en tête, s'il est indiqué en tête de l'ouvrage, et entre parenthèses, après l'éventuel nouveau titre, s'il n'est indiqué que dans le corps de l'ouvrage.

Il commence par "In" si la nouvelle référence est un chapitre d'un ouvrage collectif.

Il se termine par la mention (*résumé*) ou (*version condensée*), si le nouvel ouvrage est résumé par rapport au précédent, et par la mention (*version développée*), si le nouvel ouvrage comporte des développements par rapport au précédent.

Le champ NOUVREF ne saurait influencer un des champs précédents. Ceux-ci concernent en effet tous la même référence, antérieure à celle figurant dans NOUVREF. Par exemple, le champ CONG n'est pas rempli à nouveau postérieurement si NOUVREF concerne un séminaire. Le champ NOUVREF peut en revanche influencer trois des champs suivants : LAN, PR et SITU.

S'il y a plusieurs "nouvelles références", celles-ci sont séparées par l'abréviation "&" et placées dans l'ordre chronologique.

Ce champ se termine par un point.

Exemple : 2ème édition, 1983, 240 p. multig.

1982, 198 p.

L'état civil en République Unie du Cameroun. Situation en octobre 1979 (par D. SIEYOJO). Yaoundé : Sous-Direction des Etudes de Population, 1980ca, 16 p. multig. (résumé).

In Rapport du séminaire national sur la population, l'emploi, la formation et le développement. Yaoundé (23-26 novembre 1977). Genève : OIT, 1978ca, 116 p. multig., pp. 60-91 (version condensée) & Cah. ORSTOM Sér. Sci. Hum. (Paris), vol. XVIII, n 1, 1981-82, pp. 107-147.

In Le Nord du Cameroun. Des hommes, une région. Paris : ORSTOM, 1984, 551 p. + 3 cartes h.t., pp. 201-301. (Collection Mémoires, n 102).

Revue Science et Technique/ Science and Technology Review (Yaoundé), vol. 1, n 1, pp. 45-55.

Echanges, développement et hiérarchies dans le Bamenda pré-colonial (Cameroun). Stuttgart : Franz Steiner Verlag Wiesbaden, 1985, XIV + 323 p. (Studien zur Kulturkunde, Band 76).

19. LAN Langue(s) de l'ouvrage

Il s'agit des langues utilisées dans le texte de la référence originale ou de celle figurant dans NOUVREF, à l'exclusion des résumés. Les codes suivants (norme ISO/R 639) sont utilisées :

FR français	RU russe
EN anglais	PT portugais
DE allemand	NL hollandais
IT italien	ES espagnol

Se référer à la norme, si la langue n'est pas sur cette liste. Si tout ou partie du texte figure dans des langues différentes, celles-ci sont mentionnées par les codes séparées d'une virgule sans espacement.

Exemple : FR

FR.EN

20. CHAP Chapitre

En majuscules (voir liste).

Un seul chapitre par document.

Exemple : METHODOLOGIE.

21. SCHAP Sous-chapitre

En majuscules (voir liste commune avec les descripteurs).

Un seul sous-chapitre par document.

Exemple : ETAT CIVIL.

22. DES Descripteur(s)

Idem que pour les sous-chapitres, mais le champ DES peut contenir plusieurs descripteurs ; dans ce dernier cas, ils sont séparés par une virgule sans espacement.

Le premier des descripteurs est toujours identique au sous-chapitre.

Exemple : ETAT CIVIL,MORTALITE

METHODOLOGIE.RECENSEMENT

FECONDITE.

23. CDES Candidat(s) descripteur(s)

Il s'agit de descripteurs potentiels, ne figurant pas encore sur la liste des descripteurs. On peut donc ajouter n'importe quel candidat descripteur nouveau, qui paraît adéquat, si celui-ci ne figure pas déjà sur la liste. Il faut consulter cette dernière pour mettre en priorité un terme déjà retenu et pour conserver rigoureusement la même orthographe.

S'écrit en majuscules. S'il y a plusieurs candidats descripteurs, ceux-ci sont séparés par une virgule sans espacement.

Exemple : MORTINATALITE

AGE

MST.SIDA.

24. PROV Province(s) du Cameroun

Il s'agit des provinces concernées par le document. Les abréviations suivantes sont utilisées :

AD pour Adamaoua	NW pour Nord-Ouest
CE pour Centre	OU pour Ouest
EN pour Extrême-Nord	SU pour Sud
LT pour Littoral	SW pour Sud-Ouest
ES pour Est	NO pour Nord
CA pour Cameroun (ensemble du pays ou province non déterminée).	

Les abréviations sont séparées par une virgule sans espace-ment, quand il y en a plusieurs.

L'abréviation CA ne peut figurer en même temps qu'une des provinces. cependant, si certains lieux spécifiques apparaissent, ceux-ci sont mentionnés dans le champ LIEUX.

Exemple : LT

EN, NO, AD

CE, SU.

25. LIEUX Autres lieux géographiques

Il s'agit de tous les autres lieux géographiques du Cameroun (en dehors des provinces), concernés par le document. Les noms de lieux étrangers au Cameroun ne sont pas relevés.

S'écrit en majuscules. S'il y a plusieurs noms de lieux, ceux-ci sont séparés par une virgule sans espacement. Les noms sont écrits "à la française" pour la partie francophone du pays et "à l'anglaise" pour la partie anglophone.

Pour les "villes" et les "villages", le nom figure seul. Pour les autres lieux, la nature du lieu est précisée à la suite, après un point sans espacement. La nature du lieu est écrite en français quand le terme approprié existe et en anglais dans le cas contraire.

C'est la dénomination originelle des unités administratives, c'est à dire leur dénomination au moment de la publication, qui est relevée, même si cette dénomination a changé. Si le nom de l'unité elle-même a changé, l'ancien nom est mis dans un premier "article" et le nouveau nom dans un autre "article", de manière que cette unité puisse être sélectionnée avec le nouveau nom. De même, en cas d'éclatement d'une unité administrative, les nouveaux noms sont mis dans autant de nouveaux "articles".

Pour les localités inférieures à la sous-préfecture (ou au chef-lieu de district) et pour les unités rurales inférieures à l'arrondissement (ou au district), on précise dans un deuxième "article" l'arrondissement (ou le district) concerné. Pour les quartiers des villes, on précise le nom de la ville.

Tous les espacements sont supprimés à l'intérieur du champ LIEUX pour permettre une sélection correcte et remplacés par des tirets, sauf toutefois quand il y a des chiffres.

Exemple : Cas d'une "ville", ou de toute localité ayant au moins des fonctions de chef-lieu de district

DOUALA
YAOUNDE
MAROUA
LOLODORF
REY-BOUBA
MOUTOUROUA

Cas d'un "village", ou de toute localité inférieure à un chef-lieu de district

ZALOM I, AKONOLINGA, ARRONDISSEMENT

Cas du quartier d'une ville

NEW-BELL, QUARTIER, DOUALA
BRIQUETERIE, QUARTIER, YAOUNDE

Cas d'une unité administrative au moins égale au district

FARO-ET-DEO, DEPARTEMENT
BOUMBA-NGOKO, DEPARTEMENT
JAKIRI, ARRONDISSEMENT
MOUTOUROUA, DISTRICT

Cas d'une unité administrative inférieure au district

OUPAY, MASSIF, KOZA, ARRONDISSEMENT
TCHERE, CANTON, MERI, ARRONDISSEMENT

Cas d'une unité administrative ayant changé de nature

LOLODORF, CIRCONSCRIPTION

Cas d'une unité administrative ayant changé de nom

BAMOUN, DEPARTEMENT, NOUN, DEPARTEMENT

Cas d'une unité administrative ayant éclaté puis changé de nom

MARGUI-WANDALA, DEPARTEMENT, MAYO-
TSANAGA, DEPARTEMENT, MAYO-SAVA, DEPARTEMENT.

Cas d'une autre unité

BAMBOUTOS, MONTS
BAMENDA, ARCHIDIOCESE
BAMILEKE, PAYS
BANYO, LAMIDAT
BORNOU, EMPIRE
BORNOU, SULTANAT
KUMBO, DIOCESE
MANDARA, ROYAUME
MBO, PLAINE
NKAM, VALLEE
TCHAD, BASSIN
WOURI, ESTUAIRE
WOURI, REGION.

26. ETH Ethnie(s)

S'écrit en majuscules, d'après la liste des ethnies.

Les noms sont séparés par une virgule sans espacement, quand il y en a plusieurs.

Exemple : BAMILEKE

MOFOU, GUEMJECK, ZOULGO.

27. CETH Candidat(s) ethnie(s)

Il s'agit des autres groupes de population ne figurant pas (ou pas encore) sur la liste des ethnies.

Lorsqu'il s'agit d'un groupe identifié, la nature du groupe est précisée à la suite, après un point sans espacement.

Ce champ s'écrit comme le champ ETH.

Exemple : NDAKA, BEKE

MUKULEHE. CLAN.

28. DATE Date de création du document dans la base

La date est exprimée en six chiffres représentant, dans l'ordre, l'année, le mois et le jour.

Il faut noter que la date peut être antérieure à l'année de publication. Le cas se présente si un document a été entré dans la base sous une forme provisoire, puis publié postérieurement. La référence a ensuite été corrigée selon la forme définitive, mais la date de création du document ne change pas.

Exemple : 831007 (pour le 7 octobre 1983).

29. PR Producteur(s) du document dans la base

Le producteur est le responsable de l'entrée d'un document dans la base. Il est représenté par un numéro (voir liste).

Il peut y avoir plusieurs producteurs : un pour la référence principale et d'autres pour les nouvelles références du champ NOUVREF. En revanche, des corrections orthographiques ou de simples ajouts dans une référence existante n'entraînent pas l'ajout d'un producteur supplémentaire.

Quand il y a plusieurs producteurs, les divers chiffres sont séparés par une virgule, sans espacement.

Exemple : 1

1.2.

30. SITU Situation physique de l'ouvrage

Il s'agit du lieu où l'ouvrage peut être physiquement consulté (voir liste).

Ce champ est rempli en fonction de l'une quelconque des références correspondant soit aux champs AU à COLL, soit au champ NOUVREF, ou de l'un quelconque des volumes, si plusieurs volumes sont référencés dans le même "document". Cela veut dire qu'un organisme est mentionné dans le champ SITU s'il possède une des références ou un des volumes cités.

S'écrit en majuscules. Si un ouvrage a été repéré dans plusieurs lieux, ceux-ci sont séparés par une virgule sans espacement.

Exemple : IFORD.ISH
ORSTOM.

31. CONTENU Contenu

Il s'agit d'un résumé de l'ouvrage, d'un commentaire, d'une analyse critique ou d'une précision.

Ce champ peut contenir notamment le nom d'un auteur connu, mais qui ne figure pas sur le document. On peut aussi signaler ici le caractère de diplôme universitaire d'un travail donné, si cette précision ne figure pas sur l'ouvrage en question.

Si l'auteur du "contenu" est aussi l'auteur de l'ouvrage, tout le champ est écrit entre guillemets (cas des résumés d'auteur). dans le cas contraire, les initiales de l'auteur du "contenu" figurent à la fin du champ entre guillemets (voir liste des auteurs des "contenus").

Le champ CONTENU est rédigé normalement, en écriture "riche" (avec majuscules, minuscules, espacements et ponctuation). Il se termine donc par un point.

Exemple : Contient un paragraphe sur la composition ethnique et la constitution territoriale (pp. 13-14) (P.G.).

LOGICIELS D'APPLICATION A LA GESTION DE LA BIBLIOTHEQUE

Christiane Humbel
(informatique documentaire)
(Unité centrale de Documentation)

Centre ORSTOM de Bondy
70/74 route d'Aulnay
93140 Bondy (France)

RESUME - Présentation de deux logiciels programmés avec le logiciel LOGOTEL de l'environnement TEXTO : gestion des commandes d'ouvrages et gestion des prêts.

1. GESTION DES COMMANDES D'OUVRAGES

Le menu propose :

- la saisie d'une nouvelle commande ;
- la liste des commandes en cours ;
- une correction ou une saisie de nouveaux champs sur une commande en cours ;
- la liste des commandes reçues ;
- la liste des fournisseurs ;
- le point financier sur le montant des factures des commandes reçues ;
- le point financier sur les sommes engagées par les commandes en cours ;
- une saisie (simplifiée) des commandes dont les factures sont "*liquidées*" par la comptabilité et un point financier sur le montant de ces factures ;
- une recherche sur les commandes en cours :
 - . par le titre de l'ouvrage
 - . par l'auteur
 - . par le numéro de commande
 - . par le fournisseur
 - . par le nom du demandeur
- une recherche sur les commandes reçues, par le titre de l'ouvrage ou par l'auteur ;
- l'envoi d'une réclamation envers un fournisseur.

La saisie d'une nouvelle commande correspond au remplissage d'un bon de commande. Les champs proposés sont les suivants :

- type de crédits
- numéro de commande
- date de commande (automatiquement la date du jour)
- nom du demandeur
- titre et auteur de l'ouvrage commandé
- collection (éventuellement)
- nom de l'éditeur
- lieu d'édition
- date de parution
- nombre de volumes commandés
- nombre d'exemplaires commandés
- prix prévu
- coordonnées du fournisseur.

A ce niveau de saisie, seul le champ numéro de commande est obligatoire : l'opérateur peut y revenir en mode correction (texto LW) correspondant à un des choix du menu.

On demande également d'entrer une date de livraison prévisible (à + 1 ou + 2 mois de la date du jour). Le test de la date du jour par rapport à cette date de livraison permet de vérifier s'il y a lieu d'envoyer une réclamation à un fournisseur.

Au fur et à mesure du traitement de la commande, d'autres champs sont remplis (accès par le choix "*correction d'une commande*") :

- accusé de réception et délai de livraison prévu (indications envoyées par le fournisseur) ;
- date de réception de l'ouvrage commandé ;
- date de transmission au pôle documentaire pour l'indexation ;
- date de réception et numéro de la facture ;
- date de passation au service comptabilité ;
- quand tous les champs sont remplis, transfert de la commande dans le fichier des commandes reçues ;
- suivant les cas : copie automatique dans le fichier des commandes annulées ou dans celui des ouvrages épuisés pour ne pas encombrer inutilement le fichier.

On peut lister les commandes en cours par type de crédits ou par nom de fournisseur. Cette application fonctionne à BONDY.

2. GESTION DES PRETS

Le menu propose :

- l'enregistrement d'un nouveau lecteur ;
- le prêt d'un ouvrage ;
- le renouvellement du prêt d'un ouvrage ;
- le retour d'un ouvrage ;
- le prêt d'un périodique ;
- le renouvellement du prêt d'un périodique ;
- le retour d'un périodique ;
- la vérification de la date de retour des ouvrages et des périodiques empruntés pour lancer d'éventuelles réclamations auprès des lecteurs en retard ;
- une recherche sur les ouvrages et les périodiques empruntés :

- . par le numéro d'inventaire pour les ouvrages
- . par la cote pour les périodiques
- . par l'auteur ou le titre
- . par le nom du demandeur

- le passage au programme de gestion des commandes pour lister par exemple, les commandes en cours pour un éventuel lecteur demandeur. A chaque lecteur est attribué un code (il n'y a pas de carte de lecteur).

Le fichier des périodiques courants est relié à celui des périodiques empruntés pour éviter une saisie fastidieuse du titre, souvent très long, de ces périodiques.

La bibliothécaire entre la date de retour prévue en fonction de la catégorie de lecteur (étudiant, stagiaire, personnel ORSTOM Bondy ou extérieur) et de la longueur du prêt (en général quinze jours pour les prêts à court terme, les plus fréquents, mais il y a de des prêts à long terme et des ouvrages affectés à une unité de recherche, donc à très long terme).

Ce programme peut être adapté pour toutes les unités soucieuses de gérer l'entrée et la sortie de leurs documents personnels : la saisie des références ne prend pas plus de deux minutes, on peut retrouver rapidement "qui a emprunté quoi ?" soit avec l'option "recherche", soit avec l'option "réclamations". Il fonctionne sous Unix et sous DOS et il suffit de disposer du logiciel TEXTO.

HORIZON, BASE BIBLIOGRAPHIQUE ORSTOM : PRESENTATION

Jean-Paul Roux-Fouillet
(informatique documentaire)
(Unité centrale de Documentation)

Centre ORSTOM de Bondy
70/74 route d'Aulnay
93140 Bondy (France)

RESUME - La création de la base bibliographique ORSTOM répond à plusieurs objectifs :

- améliorer et multiplier les services documentaires rendus aux chercheurs ORSTOM ;
- valoriser et diffuser les travaux des chercheurs ORSTOM ;
- permettre un accès décentralisé aux fonds ORSTOM, notamment pour les centres et missions.

La base HORIZON se définit comme une base de références de l'ensemble des publications des auteurs ORSTOM.

Organisation générale

Lancée le 1er septembre 1986, l'opération HORIZON a démarré autour d'un schéma proposé à l'ensemble des structures ORSTOM :

- un logiciel unique, TEXTO (société CHEMDATA), choisi en raison de sa portabilité (PC, UNIX, etc.) ;
- un bordereau d'analyse documentaire unique ;
- des matériels compatibles (PC-DOS pour les centres et missions) ;
- des instruments de travail commun : manuels, vocabulaires, etc..

Le tout s'articulant autour d'un centre, l'Unité de Documentation de Bondy, équipée en 1986 d'une unité centrale : MICRO-MEGA 32 SXT (ALCATEL), sous UNIX, apte à fusionner les fichiers saisis sur place et dans les centres et missions.

Actuellement il y a seize implantations de TEXTO à l'ORSTOM, dont les principaux centres (Bondy, Montpellier, La Fayette, Brest, Nouméa, Dakar, etc.). Plusieurs milliers de références ont été saisies.

Perspectives

Les années 1987 et 1988 seront celles :

- du traitement du rétrospectif (25 000 références) ;
- de l'adoption d'un "rythme de croisière" pour le courant ;
- de la mise en place de produits documentaires nouveaux (bulletin bibliographique, diffusion sur profil, etc.) ;
- de développements informatiques : logiciel de création du bulletin bibliographique, logiciel de gestion de bibliothèque, logiciels de reformatage pour le chargement des références ORSTOM au CIRAD (et AGRIS) et dans IBISCUS, etc. ;
- d'un rééquipement matériel du "centre" : renforcement des capacités informatiques et connexion au "réseau ORSTOM" (Ethernet).

1989 devrait être l'année du chargement de la base sur un serveur, afin de la rendre accessible à nos partenaires scientifiques, ainsi qu'aux centres et missions (en fonction des réseaux internationaux).

Avec HORIZON, l'ORSTOM se trouve enfin à même d'engager des opérations de coopération dans le domaine documentaire, à l'exemple du Cameroun.

1. OBJECTIFS DE LA BASE HORIZON, PRODUITS ATTENDUS

1.1. Justification, objectifs

La base *HORIZON* est une base de données des références bibliographiques des travaux des chercheurs ORSTOM (ou associés à l'ORSTOM dans le cadre d'un programme de recherche).

Ses objectifs sont définis dans le cadre du Schéma directeur informatique documentaire, lui-même partie intégrante du schéma directeur informatique ORSTOM.

Elle est destinée à :

- améliorer la gestion et le traitement documentaire des travaux ORSTOM ;
- valoriser à l'extérieur les publications des chercheurs ORSTOM ;
- diffuser plus largement les résultats des programmes de recherche ;
- permettre à l'ORSTOM de répondre dans de meilleures conditions à des demandes de reconstitution de fonds documentaires et d'ingénierie documentaire (actions de coopération) ;
- insérer l'ORSTOM dans un réseau francophone de communication et d'échange d'IST.

HORIZON s'est constituée depuis septembre 1986 autour d'un équipement central à Bondy, mais elle associe dans un réseau les Centres de documentation spécialisés (CDS) de Bondy (Formations Superficielles), Montpellier (Sciences de la Terre, Santé, Hydrologie) et le CEDID (Sciences économiques et Sociales). Les principaux centres et missions ORSTOM contribuent à son alimentation.

HORIZON développe les logiciels nécessaires pour l'alimentation des bases bibliographiques des partenaires de l'ORSTOM (CIRAD, IBISCUS) et des bases internationales spécialisées (AGRIS).

Elle participe à des actions de coopération en matière d'ingénierie documentaire : informatisation des documentations du MESRES camerounais et de ses instituts de recherche notamment.

En 1989, l'objectif est d'avoir informatisé l'ensemble des publications ORSTOM depuis 1947 : des accès en ligne seront alors ouverts et l'accès aux documents sera assuré par le recours à un fonds de microfiches (dont la réalisation complète prendra plus de 2 ans).

1.2. Produits attendus :

- Bulletin bibliographique ORSTOM, extrait de la base, dont le numéro 00 est sorti en août 1987. L'année 1988 sera celle de la sortie du numéro 1.

- "*Diffusion sélective de l'information*", à travers la fourniture de produits documentaires sur profil : bulletins bibliographiques et fichiers informatisés pour les centres et missions d'une part, pour nos partenaires du Sud d'autre part, comme cela s'est fait pour le Cameroun ; il s'agit en quelque sorte de mettre au catalogue ORSTOM des "*produits intégrés*" de recherche documentaire.

- Accès à la base depuis les centres et missions ORSTOM, en fonction de l'état des réseaux internationaux de transmission de données.

- Alimentation des bases internationales et françaises spécialisées.

2. ORGANISATION GENERALE

2.1. Un réseau documentaire

HORIZON se présente comme un *réseau documentaire* (au sens organisationnel du terme) au sein de l'ORSTOM. Ce réseau comprend trois niveaux :

- le centre de la base, sis à Bondy ; l'unité "*HORIZON*" assure l'administration générale de la base, le contrôle des en-

trées/sorties, la coordination des travaux documentaires liés à la base (vocabulaires, etc.), les développements informatiques, l'harmonisation des matériels et des formats. Elle fournit les équipements mi-lourds (en liaison avec la Mission Technique Informatique) et elle incite les centres et missions à "*adhérer*" au réseau HORIZON (aide à l'acquisition des logiciels et matériels, aide à la mise en oeuvre, notamment par des formations, assistance téléphonique). Enfin elle développe une politique active de coopération là où l'ORSTOM est organisé en mission et chez nos partenaires scientifiques des pays du sud ;

- trois centres de documentation spécialisés : le CDS de Bondy (Formations Superficielles), le CDS de Montpellier (Sciences de la Plante, Hydrologie, Santé/Nutrition), le CEDID de la rue La Fayette (Sciences Economiques et Sociales). Chaque CDS assure toutes les fonctions d'un centre de documentation dans ses disciplines, alimente et interroge la base, participe à la mise à jour des outils documentaires ;

- les centres et missions outre-mer qui adhèrent au réseau (liste en annexe). Une "*charte*" (en cours de rédaction) définit les relations avec le centre : fourniture dans un format commun des notices des documents édités localement (disquettes), réception des fichiers des documents ORSTOM édités centralement (il faut alors rajouter la cote locale).

2.2. Les règles

Le fonctionnement du réseau repose sur une série de règles et formats communs : format de saisie (bordereau ORSTOM, présenté en annexe), format d'interrogation simplifié, vocabulaires communs (disponibles dans la plupart des disciplines), instruments de travail communs (manuel d'interrogation et de saisie, plan de classement).

Les bibliographies personnelles des chercheurs peuvent être saisies sous TEXTO ; cependant le coût du logiciel (7500 F) et le caractère relativement complexe des formats de saisie constituent des freins.

Actuellement HORIZON travaille sur un format de saisie très simplifié et souhaite pouvoir fournir le logiciel à prix avantageux : mais ceci est un problème de budget.

2.3. Microthèque

Enfin une microthèque est associée à HORIZON, pour permettre l'accès aux documents.

3. MATERIELS ET LOGICIELS

3.1. Logiciel

Le logiciel est *TEXTO* : choisi en raison de sa portabilité quasi-universelle dans une version identique, il permet une grande souplesse dans l'évolution d'une application et dans l'adaptation à des structures complexes comme celles de l'ORSTOM. Il a été adopté par beaucoup de nos partenaires : CIRAD, INRA, INSERM, Ministère de la Coopération, etc..

TEXTO se présente comme une boîte à outils : l'application est à écrire, notamment en utilisant le langage de programmation associé (*LOGOTEL*). Il est particulièrement performant pour les applications documentaires de taille moyenne (ce qui est le cas de l'ORSTOM). Ses deux grosses qualités : ses capacités de tri et son format variable qui économise la place disque (ces deux qualités le différencient de *DBASE*). Ses défauts : pas de calcul sauf l'addition, une certaine lourdeur à la saisie (mais la dernière version, 4.1, présente une vraie saisie pleine page).

Outre *TEXTO/LOGOTEL*, *HORIZON* utilise *TEXLOAD*, logiciel de "téléchargement" et de reformatage sous *TEXTO* des fichiers capturés depuis les serveurs *DIALOG*, *QUESTEL* et *ESA* ; joint à la carte *KORTEX*, *TEXLOAD* est un instrument de travail très intéressant (il permet en outre de reformater sous *TEXTO* les fichiers en format *MICROQUESTEL*, par exemple les disquettes "*PASCAL DSI*" du CNRS).

Une application bureautique tourne enfin à Bondy, sous *MULTITEXT* : elle permet notamment de sortir des bulletins bibliographiques.

3.2. Matériels

Les matériels préconisés au sein du réseau *HORIZON* sont des micros DOS compatibles PC. Au niveau central, on rentre dans le cadre du schéma directeur informatique ORSTOM.

Les matériels utilisés actuellement sont les suivants :

- à Bondy un *MICROMEGA 32 SX 70T* avec 2 Mo de mémoire centrale, 70 Mo de disque ; système d'exploitation : *MIMOS 2.1*, "une" version 7 d'*UNIX*. 11 postes connectés, dont 2 micros DOS (*Micral 30*) ; logiciel de transfert de fichier et d'émulation : *MULTIWORKS (PCWORKS)*.

- dans les CDS et centres, des micros divers ; actuellement *TEXTO* est implanté à l'ORSTOM sur des *IBM AT* et *XT*, sur *Micral 30*, sur *Olivetti M24*, *Cordata* etc..

3.3. Développements

Plusieurs logiciels ont été développés ou sont en cours de développement :

- logiciels liés à la gestion de l'application ORSTOM : fusion de fichiers, menus, etc. ;
- logiciels de gestion : gestion des prêts à la bibliothèque, gestion des commandes (ces deux logiciels font l'objet d'une présentation séparée), logiciel de bulletinage ;
- logiciels de reformatage vers le CIRAD (puis AGRIS) et IBISCUS ;
- logiciel d'édition du bulletin bibliographique.

4. PERSPECTIVES 1988-1989

Traitement du "rétrospectif" ORSTOM (24000 références) grâce à un financement FAC (2MF) : traitement informatique et complément de traitement documentaire d'ici fin 1989 ; microfichage des mêmes documents, avec un délai plus long. Dans cette perspective, trois marchés devront être lancés fin 1987 et début 1988 : le premier sous-traitera le complément de traitement documentaire et la saisie (fin 1987), le deuxième permettra le rééquipement de l'atelier de micrographie (passage à la partition 24x et modernisation) (fin 1987), le troisième fera appel à un façonnier pour une partie du microfichage.

Extension du réseau à de nouveaux centres ou missions : Ouagadougou et Yaoundé en 1988, deux autres centres ou missions en 1989 (ce qui implique la fourniture du logiciel, l'aide à l'acquisition du matériel, une formation et une mission d'implantation).

Programme d'équipement informatique :

- acquisition d'une nouvelle unité centrale à Bondy en 1988 pour renforcer la première (notamment afin de développer dans de bonnes conditions les applications bureautiques liées à l'informatisation de la gestion) et pour entrer totalement dans le cadre du schéma directeur informatique (connexion ETHERNET) ;
- équipement informatique "mi-lourd" du CDS de Montpellier en 1988 (avec connexion ETHERNET) ;
- acquisition d'un équipement multiposte au CEDID dès 1988.

Poursuite des *développements informatiques* nécessaires pour alimenter les bases extérieures (IBISCUS, CIRAD, AGRIS, URBAMET), pour permettre un meilleur confort et un contrôle de la saisie (programme de saisie assistée), et pour automatiser la gestion (adaptation des logiciels de gestion des prêts, des commandes et du bulletinage réalisés en 1987).

Mise en place progressive d'accès en ligne à Horizon : en 1988 depuis la rue La Fayette et Montpellier (via TRANSPAC et le réseau ORSTOM) ; en 1989 en étudiant la possibilité d'un portage sur serveur externe.

Poursuite d'une politique active en matière de coopération : opérations de reconstitution de fonds documentaires, de formation et de mise à disposition de "*structures intégrées de recherche documentaire*" au Tchad, à Madagascar et en Guinée-Konakry en 1988 et 1989 (en plus de la poursuite des opérations Cameroun et Institut du Sahel).

Insertion d'HORIZON dans le réseau francophone de communication et d'échange d'IST qui tente de se mettre en place actuellement (cf Sommet des chefs d'Etat francophones en septembre 1987).

Etude de faisabilité d'un CD-ROM ORSTOM (1989), permettant la mise à disposition des centres, des missions et de nos partenaires, de l'ensemble de la base HORIZON, avec mise à jour annuelle, sur micro-ordinateur.

ANNEXE 1 :LISTE DES FICHIERS DISPONIBLES A BONDY ET AU CEDID

- + Le fichier central HORIZON : 600 réf.
- + Le fichier CEDID (rue La Fayette) : 2600 réf.
- + Le fichier Cameroun : 2600 réf.
- + Le fichier POPCAM : 2400 réf.
- + Le fichier Nouméa : 500 réf.
- + Le fichier Dakar : 400 réf.
- + Le fichier Hydro Montpellier : 400 réf.
- + Le fichier Océano Brest (doc ORSTOM Nossi Be) : 500 réf.

ANNEXE 2 :LISTE DES IMPLANTATIONS TEXTO AU SEIN DE L'ORSTOM

Le réseau HORIZON comprend actuellement, outre les CDES de Bondy et Montpellier et le CEDID, les implantations suivantes, qui ont toutes créées au moins des fichiers de gestion documentaire locale sous TEXTO (mais nous manquons d'informations sur Cayenne et Lomé) :

- + ATOB Télédétection (Bondy)
- + UR 104 Pédologie (Bondy) ;
- + Département D rue La Fayette ;
- + Antenne auprès du Muséum d'Histoire Naturelle de Paris ;
- + Antenne ORSTOM auprès de l'IFREMER (Brest) ;
- + Atelier d'Hydrologie Montpellier ;
- + Centre ORSTOM de Tunis ;
- + Centre ORSTOM de Nouméa ;
- + Centre ORSTOM de Dakar ;
- + Centre ORSTOM de Lomé ;
- + Centre ORSTOM de Cayenne ;
- + UR 202 d'Adiopodoumé ;
- + CEPED (Paris).

prévisions 1988:

- + Mission ORSTOM à Yaoundé ;
- + Centre ORSTOM de Ouagadougou ;
- + Mission ORSTOM à Fort de France ;
- + Mission ORSTOM au Vanuatu.

HORIZON Base bibliographique ORSTOM

1-88

NUM _____ DA _____
 aa mm jj
INV1 _____; _____ INV2 _____; _____ MIC M
NBEX1 _____ NBEX2 _____
COTE1 _____ COTE2 _____

NIVBIB M A S TYPBIB (forme physique du document)
 B Monographie J Article périod.
 C Norme R Rapport officiel
 G Carte/Atlas/Notice

INDBIB (type de document et éléments du contenu) BASES EXT
 K Congrès Z Bibliographie I IBISCUS
 L Glossaire/Encyc. R Synth. bibliogr. A AGRIS
 Q Répertoire/Ann. Y Cartes AS ASFA
 N Tableaux num. O Rapport d'activité
 U Thèse, diplôme V Multigr/Non conv.
 E Résumés

LAN _____; _____; _____; _____; _____ ISBN _____
LANR _____; _____; _____; _____; _____

NIV S T V DIFF 1.- Pas d'accès possible
VERSION Pa. Pr. 2.- Consultation à l'ORSTOM
CONFID 0 1 4.- Diffusion payante auprès de l'ORSTOM
BULL1 1 5.- Diffusion payante sous formes mic. ou
 photocop.
 6.- Prêt possible

A U T E U R**Document générique**AUM

AMM /1 _____ /2 _____
 /3 _____ /4 _____ /5 _____
AFM /1ORSTOM/2Institut Français de Recherche Scientifique pour le Développe-
 ment en Coopération/3 _____ /4 _____ /5 _____

Document spécifiqueAUA

AMA /1 _____ /2 _____
 /3 _____ /4 _____ /5 _____
 /1 _____ /2 _____
 /3 _____ /4 _____ /5 _____
AFA /1ORSTOM/2Institut Français de Recherche Scientifique pour le Développe-
 ment en Coopération/3 _____ /4 _____ /5 _____
 /1ORSTOM/2Institut Français de Recherche Scientifique pour le Développe-
 ment en Coopération/3 _____ /4 _____ /5 _____

TITRE
Document génériqueTITORMNOEDITIFRAMTITANM**Document spécifique**TITORATIFRAATITANA

CONGRES
NUMCGNOMCGLOGCPAYSCG ___DACG ___ / ___ / ___
aaaa mm jj

DIPLOME
DIPLUNIVDADIPL ___ / ___ / ___
aaaa mm jjNODIPL

OPERAT
OPERAT

NOBRA
NOBRA

COLLATION

ILL ___ BIBLIOGR ___ CARTE ___:1/___; ___:1/___; ___:1/___
TABL ___ GRAPH ___ PHOTO ___ MATDAC ___ DICT ___

LED (___) ED
 (___)

DP ___ / ___ / ___ VOLO ___ / ___ PAGE

SERIE

JT (___) PAGE
COLL (___)
VOL ___ NUMER ___ ISSN _____ DP ___ / ___

NOTE**DESCRIPTEURS ORSTOM**CHAP _____ ; _____DES _____ CDE _____GEO _____LOC _____**LISTES SYSTEMATIQUES**

Indiquer le nom du champ:

DESCRIPTEURS BASES EXTERIEURESDAGRIS _____DIBIS _____CHAPAGR _____

RESUME

**Session 4 : INFORMA-
TIONS GEOGRAPHIQUES**

FORBAN : UN LOGICIEL INFOGRAPHIQUE DE CARTOGRAPHIE THEMATIQUE DEVELOPPE ET PORTABLE SUR MICROORDINATEUR COMPATIBLE PC

Paul Faure et Claude Benizri
(Pédologie et Informatique)
(UR 104, Géodynamique de la Surface)

Centre ORSTOM de Bondy
70,74 Route d'Aulnay
93140 Bondy (France)

RESUME - L'inventaire et l'étude de la répartition spatiale des sols, de leurs caractéristiques, et des facteurs du milieu qui conditionnent leur différenciation, constituent une des activités majeures des pédologues. Parallèlement à l'établissement de cartes pédologiques sensu-stricto, il apparait de plus en plus nécessaire, pour visualiser les différentes constituantes de l'information contenue sur un document cartographique de base, d'établir des documents cartographiques thématiques dérivés. L'Infographie, domaine de l'informatique appliqué à la cartographie permet de dresser ce type de document thématique.

L'atelier "*Expression Graphique*" de l'UR 104 "*Géodynamique de la Surface*", Département A, s'est doté d'un équipement basé sur un micro-ordinateur, un digitaliseur et un traceur, et a développé pour ce type d'équipement une application complète et compatible : FORBAN.

OBJECTIFS ayant conduit au développement de FORBAN

La réalisation de documents thématiques met en oeuvre trois principales étapes pour saisir et traiter l'information de base, en extraire et restituer l'information thématique partielle demandée.

L'information de base se présente sous forme d'une carte et d'une légende qui portent respectivement une information de type géographique : contours et zones cartographiques, et une information de type thématique globale : contenu des unités cartographiques. Le produit attendu se présente sous forme de documents cartographiques et légendes visualisables sur écran vidéo et sur papier.

FORBAN a été développé pour effectuer de manière simple, cohérente et en séquences les traitements nécessaires.

- La fonction de saisie de l'information géographique fait intervenir une digitalisation point par point sur table à digitaliser, et sa mise en mémoire sous forme hiérarchisée d'arcs et de zones. L'information thématique globale est saisie au clavier et reliée à l'information géographique.

- La fonction de traitement a pour objectif de restructurer les deux ensembles d'information et d'en extraire à la demande les éléments constitutifs des cartes thématiques : les contours des nouvelles zones cartographiques et leurs contenus thématiques. Parallèlement ces traitements permettent la restitution des documents de base et divers calculs, en particulier celui de différentes surfaces.

- La fonction de visualisation, tant sur écran que sur traceur, permet de définir de façon variée à la demande tous les paramètres de la restitution : échelle, figurés, habillage.

CONDITIONS physiques d'exploitation de l'application

FORBAN est actuellement utilisé sur l'équipement suivant :

- micro-ordinateur IBM-PC XT, 640 Ko RAM, disque dur 10 Mo, unité de sauvegarde QUADRAM de 5 Mo, imprimante parallèle, carte graphique et écran couleurs CGA ;
- digitaliseur BENSON 6301 ;
- traceur BENSON 1625-S.

Son développement a été effectué à partir de :

- QUICKBASIC (Basic compilé) ;
- assembleur (Gestion de la mémoire) ;
- FABS+ (Gestion des fichiers) ;
- GPR 200 T (Gestion du traceur) ;
- système d'exploitation MS-DOS 2.1.

FORBAN fait intervenir une trentaine de modules, adressables à l'aide d'un menu accessible à tout utilisateur, qui correspondent à chacune des étapes des différentes opérations à réaliser pour obtenir le produit. Il ne présente pas de limitation de taille dans la perspective de saisie et de traitement de documents cartographiques courants : jusqu'à 5000 zones et 15000 arcs, sans limitation du nombre de points.

Cette application reste ouverte à d'autres extensions qui sont envisagées dans l'avenir :

- croisement et assemblage de plusieurs cartes ;
- transfert de fichiers de base pour gravure extérieure de typons en vue d'éditions imprimées ;
- transformation de systèmes de projection.

CARACTÉRISTIQUES d'exploitation de FORBAN

L'application a été développée, du fait du matériel disponible, sur un micro-ordinateur de type XT, sous système d'exploitation MS DOS et langage BASIC, qui ont entraîné leurs contraintes spécifiques : capacité mémoire, taille des fichiers et vitesse de traitement limitées. Des solutions originales y ont porté remède : utilisation d'un Basic et d'un compilateur plus performants (QUICK-BASIC), utilisation de segments mémoires, limitation du volume des données, réduction du nombre d'accès aux fichiers.

D'autres améliorations pourront être apportées : par acquisition d'une unité centrale de type AT, pour laquelle FORBAN est d'ores et déjà compatible, par acquisition d'une unité de traitement à forte capacité (30 Mo), par installation de cartes mémoires supplémentaires. Il en va de même en ce qui concerne le déroulement des opérations, saisie et traitement, qui ne peuvent actuellement pas être effectuées simultanément. Elles le pourraient aisément en présence de deux postes de travail (l'actuel XT dédié aux fonctions de saisie et un AT, plus performant, dédié aux fonctions de traitement), et du fait du caractère modulaire de FORBAN.

En conclusion, le système FORBAN constitue un outil de cartographie thématique assistée par ordinateur complet et performant. Il se caractérise enfin par une remarquable simplicité d'utilisation du fait du type de matériel très répandu sur lequel il tourne et de la particulière clarté de ses commandes.

INTRODUCTION

L'inventaire et l'étude de la répartition spatiale des sols, de leurs caractéristiques, et des facteurs du milieu qui conditionnent leur différenciation, constituent une des activités majeures des pédologues. Parallèlement à l'établissement de cartes pédologiques sensu-stricto, il apparaît de plus en plus nécessaire, pour visualiser les différentes constituantes de l'information contenue sur un document cartographique de base, d'établir des documents cartographiques thématiques dérivés. L'Infographie, domaine de l'informatique appliqué à la cartographie, permet de dresser ce type de document thématique.

Le Laboratoire "*Expression Graphique*" de l'UR 104 "*Géodynamique de la Surface*", Département A de l'ORSTOM, s'est doté d'un équipement basé sur un micro-ordinateur, un digitaliseur et un traceur, et a développé une application complète et compatible : le logiciel FORBAN.

1. OBJECTIFS DU DEVELOPPEMENT ET DE L'UTILISATION DE FORBAN

Tout document cartographique fait apparaître l'information qu'il retranscrit sous deux aspects complémentaires et indissociables :

- une *information de nature thématique* : c'est le contenu-sols de la région couverte dont l'inventaire apparaît au niveau de la légende, sous forme hiérarchisée la plus complète possible, d'unités cartographiques.

- une *information de nature géographique* : c'est la localisation sur la carte proprement-dite des unités identifiables et différenciables de ce contenu-sols, qui se traduit par un zonage de la région couverte.

La totalité de l'information pédologique caractérisée et structurée est donc portée sur les deux documents qui constituent la carte pédologique de base :

- la *légende*, qui regroupe les *unités cartographiques* et fait état des caractéristiques de leur contenu-sols, permet d'en extraire une information thématique riche et objective.

- la *carte* proprement-dite, qui fait apparaître par ses contours le découpage des *zones cartographiques* correspondant aux unités de la légende, contient la totalité du découpage thématique.

Ces deux éléments complémentaires et indissociables dans leur réalisation et leur exploitation constituent le support de l'*information pédologique de base* et la source à partir de laquelle seront extraites certaines des composantes qui figureront sur les cartes thématiques dérivées.

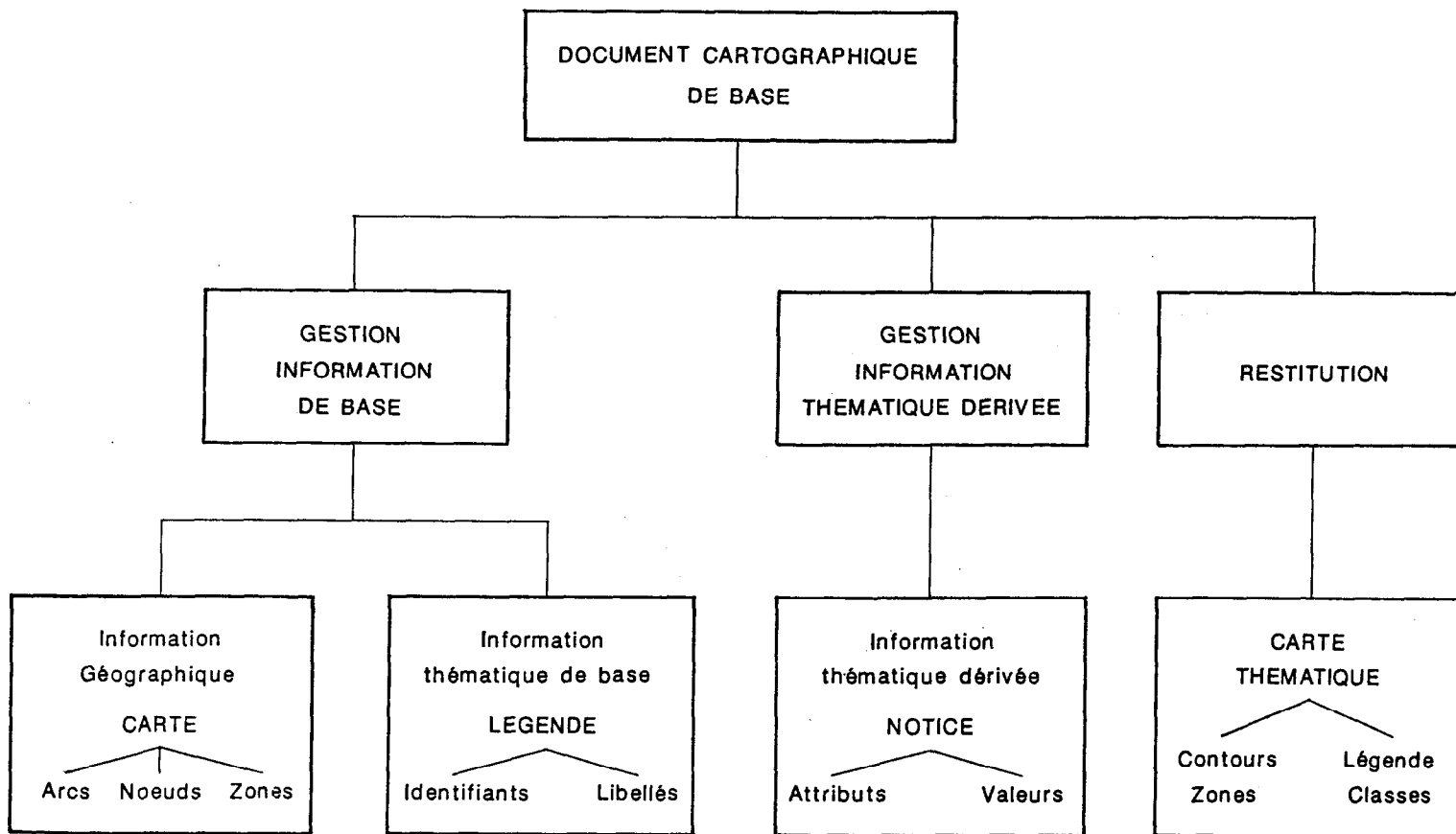
Nous allons passer en revue les différentes étapes de l'élaboration des documents thématiques à l'aide de FORBAN (Tableau 1).

1.1. Saisie de l'information cartographique de base

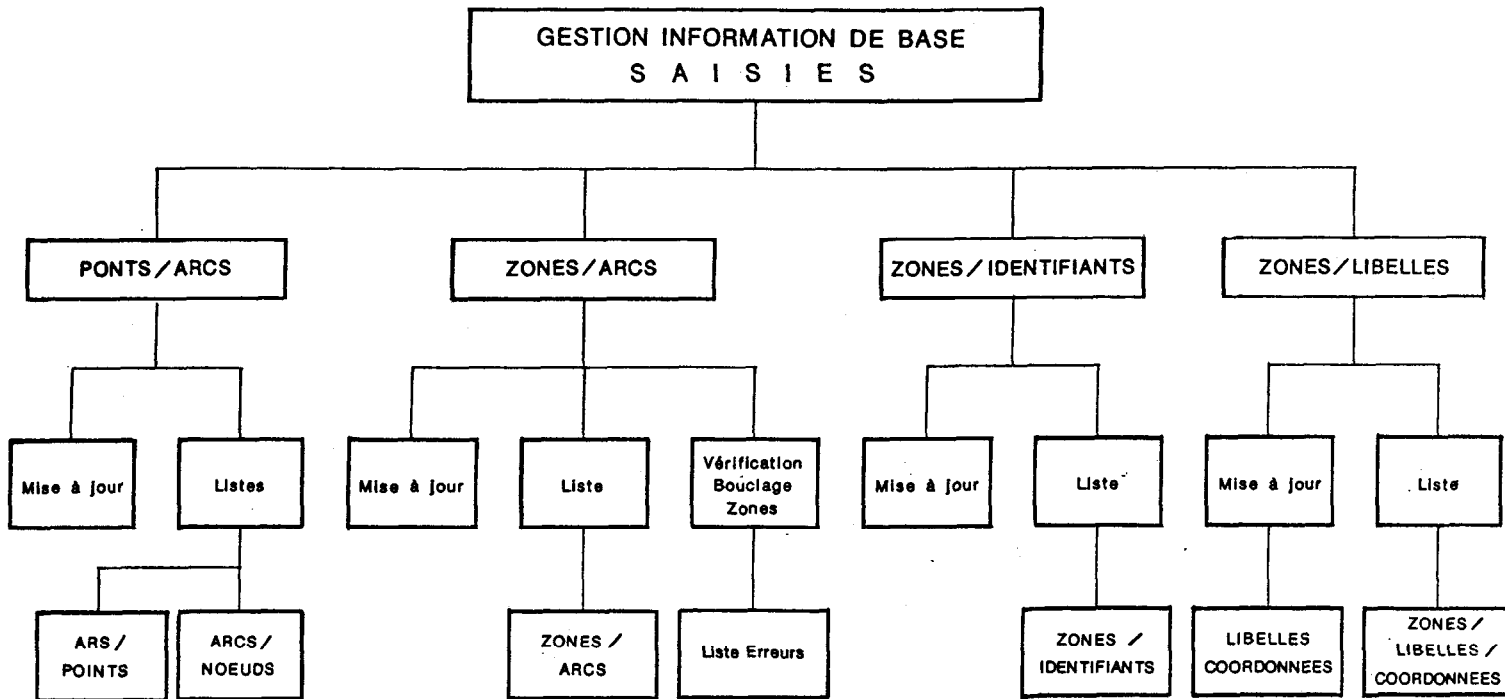
Le traitement d'une information géographiquement localisée nécessite de saisir la totalité de ses deux composantes : géographique (carte) et thématique (légende). On dispose pour ce faire de l'ordinateur et d'une table à numériser (Tableau 2).

1.1.1. Saisie de l'information géographique

Cette information se présente sur le document cartographique sous forme de *zones* que délimitent des contours. Le contour de chaque zone est constitué d'un ou plusieurs *arcs* délimités par les points de rencontre des différents arcs : les *noeuds*. Le plus souvent, un arc marque la limite entre deux zones : il leur est commun. Une opération préliminaire de numérotation de ces élé-



Tab. 1 - Organisation générale de FORBAN



Tab. 2 - Information de base : S a i s i e

ments : arcs, noeuds, zones, est effectuée sur le document de base, pour repérage au fur et à mesure de leur saisie (Figure 1).

Le principe de la saisie initiale sous FORBAN consiste ensuite à relever point par point les coordonnées de chaque arc et de ses noeuds à l'aide de la table à numériser. Ces données numériques sont transcrites et stockées en plusieurs fichiers référant les points, les arcs et les noeuds.

Une seconde saisie est effectuée au clavier de l'ordinateur. Elle recense le numéro de chaque zone et celui de ses arcs-enveloppe.

A l'issue de ces deux opérations de saisie, la totalité de l'information géographique figurant sur le document cartographique d'origine est enregistrée. Toutes ces données peuvent être listées pour contrôle.

1.12. Saisie de l'information thématique de base

Cette information se présente sur la légende sous forme de la liste des unités cartographiques et sur la carte sous forme du contenu de chaque zone du découpage. Chaque contenu de chaque zone est identifié à une des unités cartographiques de la légende d'où le terme *d'identifiant* attribué à ces unités. Il y a autant d'identifiants que d'unités cartographiques de la légende et chaque zone a son contenu caractérisé sur le plan thématique par *un seul de ces identifiants*.

Le principe de la saisie sous FORBAN de cette information consiste à constituer au clavier un *fichier identifiants/zones* qui établit la correspondance de chaque zone avec le numéro d'identifiant qui caractérise son contenu thématique de base.

Un *libellé*, chaîne de caractères, peut être également saisi pour chaque zone et positionné à l'aide du digitaliseur en vue de restitution sur traceur.

Des restitutions graphiques des contours du document de base peuvent à ce stade être déjà obtenues sur écran ou sur traceur pour vérifier la conformité des saisies avec le document d'origine.

1.13. Analyse et saisie de l'information thématique dérivée

Chaque unité cartographique (identifiant) est définie et différenciée par rapport aux autres par un ensemble de caractères qui peuvent être distingués et pris en compte séparément : ils sont dénommés *attributs*. Chacun de ces attributs peut prendre un certain nombre de *valeurs* :

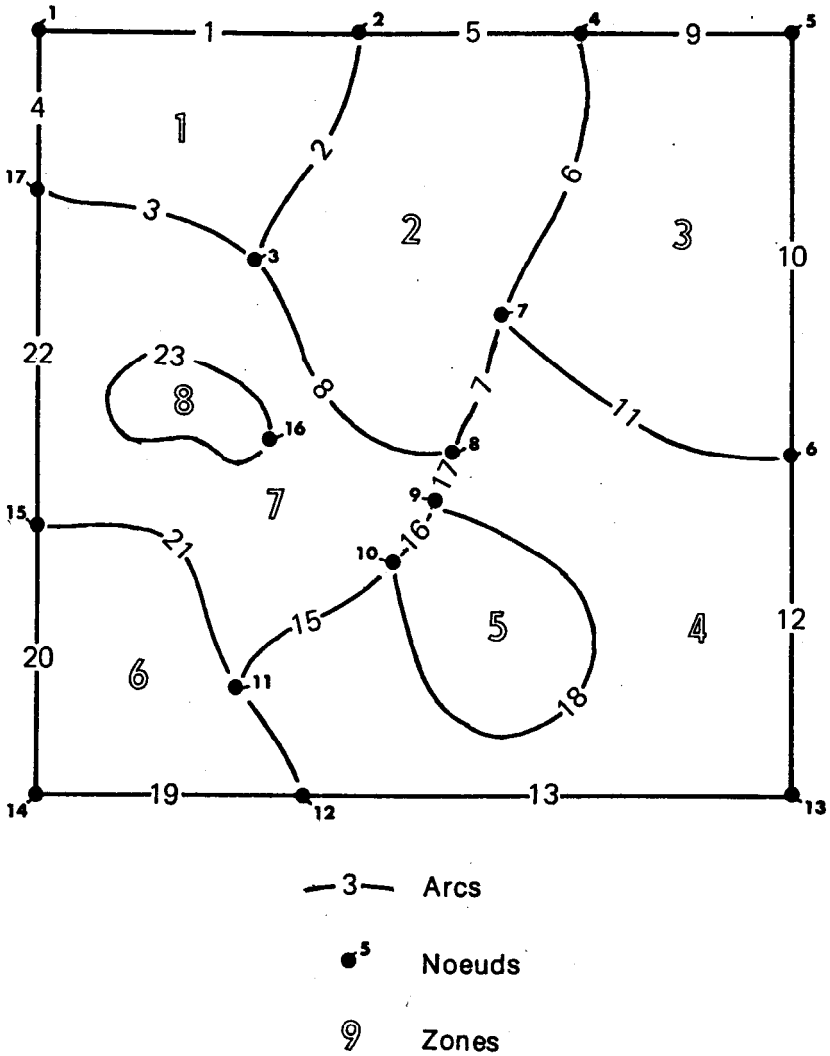


Fig. 1 - Constituantes de l'information géographique

Exemples

d'attributs : profondeur texture drainage matériau originel

Exemples	peu profond	argileuse	bon	arénacé
de valeurs :	moyen. prof.	arg.sabl.	moyen	vertique
	profond	sableuse	médiocre	isaltéritique

Cette information thématique dérivée, obtenue par analyse thématique de la légende, est saisie (Tableau 4) pour constituer :

- un *fichier valeurs/attributs* qui inventorie les différents critères (attributs) pris en compte pour caractériser et distinguer les unités cartographiques de base de façon homogène et exhaustive (texture, drainage, profondeur, etc.) et les différentes valeurs prises par ces critères ;

- un *fichier attributs/identifiants* qui stocke identifiant par identifiant la valeur prise par chaque attribut précédemment définis.

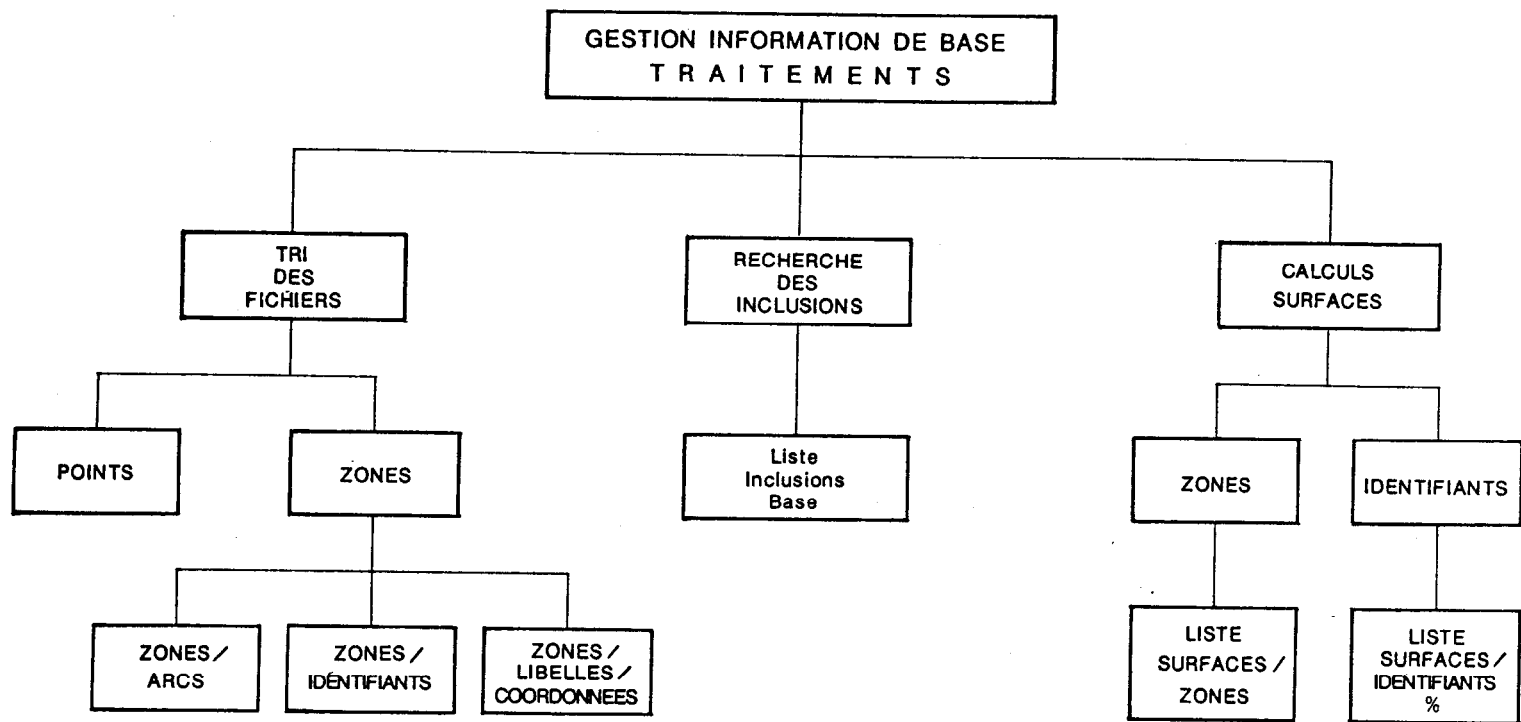
A la suite de ces nouvelles opérations de saisie, la totalité de l'information thématique analysée, divisée, hiérarchisée, classée, identifiant par identifiant et zone par zone, est stockée et peut être listée. Elle est disponible pour tout traitement (tris, calculs de surfaces, recherche des inclusions) (Tableau 3).

1.2. Traitements infographiques et documents thématiques dérivés

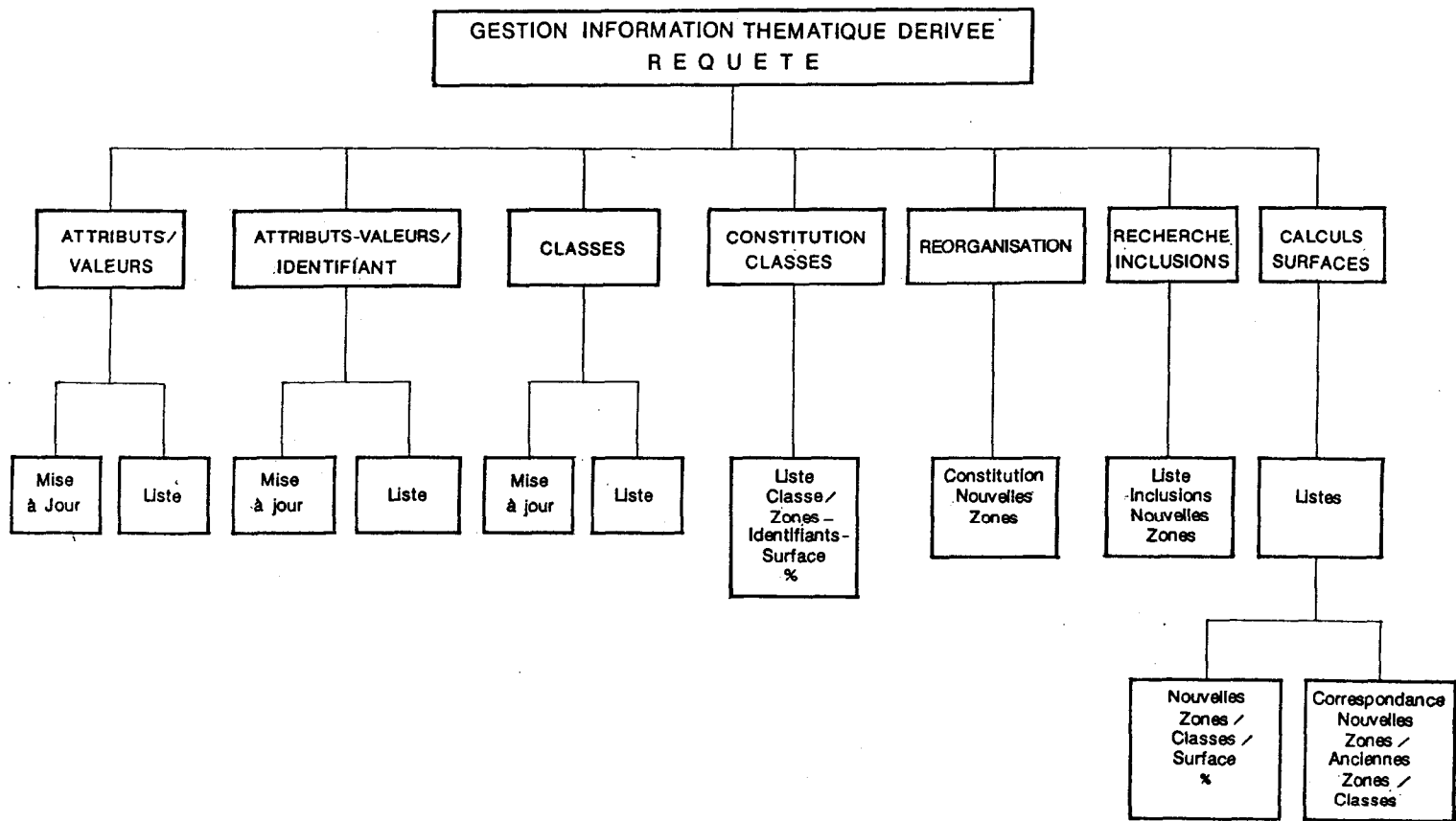
Le logiciel FORBAN permet de traiter et de croiser la totalité de l'information stockée au cours des étapes précédentes : information géographique, information thématique de base et information thématique dérivée. Il peut en extraire à la demande un ou plusieurs des éléments constitutifs pour en restituer une *image* sous forme d'un nouveau document cartographique, à nouvelles composantes, géographique (nouvelle carte), et thématique (nouvelle légende). Cette opération se déroule en trois étapes (Tableau 4).

1.21. La requête

Au cours de cette étape, on sélectionne au clavier le ou les *attributs* qui constituent le *thème* souhaité et les différentes *valeurs* de subdivision pour former les *classes* (nouvelles unités cartographiques) de la nouvelle légende. Cette légende peut comporter deux ou plusieurs classes faisant chacune intervenir un ou plusieurs attributs prenant chacun une ou plusieurs valeurs.



Tabl. 3 - Information de base : Traitements



Tabl. 4 - Information thématique dérivée : Saisie et traitement

Exemple :	Attributs	Valeurs
Classe 1	Texture	Argileuse + argilo-sableuse.
	Profondeur	Profond
	Drainage	Bon
Classe 2	Texture	Sableuse
	Profondeur	Peu prof. + moy. prof.
	Drainage	Moyen + médiocre

Le récapitulatif de la requête peut alors être listé sous une forme synthétique ; il représente la nouvelle légende et tous ses éléments constitutifs.

1.22. La constitution des classes

C'est à partir de ce niveau que FORBAN effectue les opérations les plus importantes : tris et regroupements, qui en justifient tout son développement :

- il prend en compte, classe par classe, les attributs et valeurs sélectionnés au cours de la requête ;
- il détermine et regroupe les identifiants concernés par les classes ;
- il détermine les zones correspondant aux identifiants répondant aux critères demandés pour chaque classe.

A l'issue de cette opération, une nouvelle information thématique restructurée est ainsi dégagée, tandis que pour chaque classe un sous-ensemble de zones et de leurs identifiants est regroupé et peut être listé à la demande, montrant le résultat de la requête.

1.23. La réorganisation des contours

C'est par cette opération que FORBAN crée la composante géographique du nouveau document thématique. Il réorganise les contours de la nouvelle image par maintien ou suppression d'arcs du document de base, en regroupant ou en maintenant dissociées les zones adjacentes selon qu'elles appartiennent ou non à une même classe donnée. Il crée enfin la *relation "nouvelles zones"/classes*.

A l'issue de cette opération, on dispose en mémoire, stockable sur support magnétique, d'un nouvel ensemble d'informations qui constitue l'*image thématique* : son information géographique (nouvelles zones) et son information thématique (nouvelles unités cartographiques : les classes), qui sont les données de base pour l'établissement de la *carte thématique* et de sa *légende*.

1.24. Traitements complémentaires

D'autres ensembles d'opérations sont incorporés à diverses étapes de saisie et de traitement thématique :

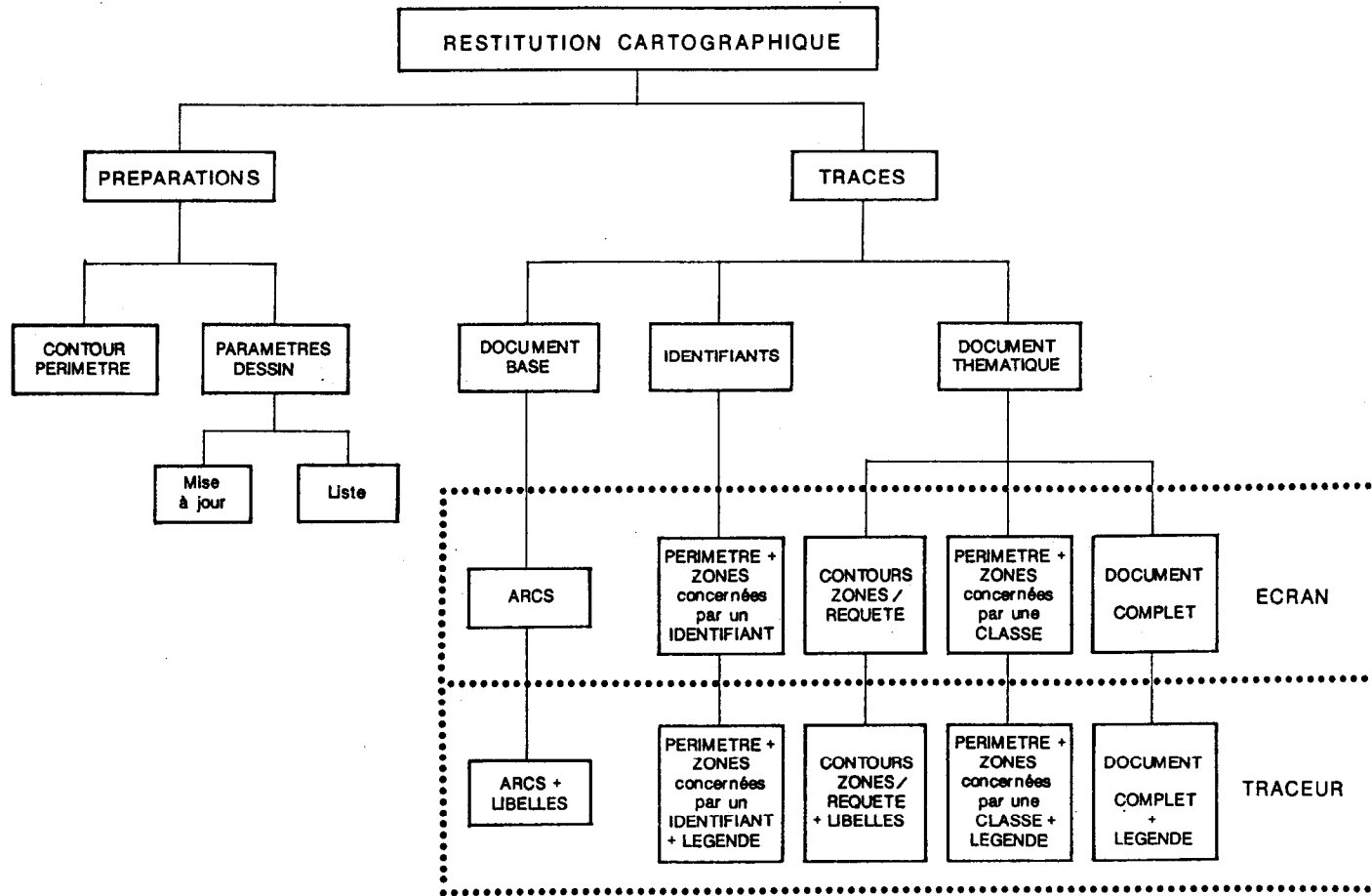
- recherches des *zones incluses* au niveau du document de base puis des documents thématiques dérivés ;
- calculs de *surfaces* des différentes zones, identifiants et classes sur tous les documents (base et thématiques), avec cumul et pourcentages ;
- détermination du *périmètre* du document de base par recherche et mise en mémoire des arcs-frontière ;
- *listes* exhaustives ou partielles de la totalité des données saisies et des divers résultats de traitement.

1.3. Restitution des documents

Une visualisation de divers documents sur écran ou sur traceur est réalisable, dans leur totalité ou en partie (fenestrage), au cours des différentes étapes de l'utilisation de FORBAN (Tableau 5) :

- restitution du document de base après saisie : contours sur écran, contours et libellés sur traceur ;
- restitution sur écran et sur traceur du périmètre et des zones de la carte de base correspondant à tel ou tel identifiant ;
- restitution sur écran et sur traceur des contours des nouvelles zones des documents thématiques, avec possibilité de figurer un libellé par zone ;
- restitution sur écran et sur traceur du périmètre et des nouvelles zones de la carte thématique correspondant à telle ou telle classe ;
- restitution sur écran et sur traceur des documents thématiques complets et de leur légende ;

La restitution à l'aide du traceur sur papier est effectuée après saisie de différents *paramètres de dessin*. Echelle de restitution, fenestrage, choix des couleurs des contours, remplissage, hachures (inclinaison, espacement), attaché à chaque élément représenté, restitution ou non des contours, sont sélectionnables au clavier. Ces paramètres sont totalement modulables et adaptables à la demande, en fonction de la nature, de la complexité du document créé, et des choix du demandeur ; ils peuvent être contrôlés par listage et modifiés pour chaque restitution.



Tabl. 5 - Traitements de restitution cartographique

2. CONDITIONS PHYSIQUES D'EXPLOITATION DE FORBAN

2.1. Description du matériel et des logiciels

FORBAN est actuellement utilisé sur l'équipement suivant :

- un micro-ordinateur IBM-PC XT possédant 640 Ko RAM, un disque dur de 10 Mo, une carte graphique CGA et un écran couleur ;

- une unité de sauvegarde QUADRAM de 5 Mo ;
- une imprimante parallèle ;
- un digitaliseur BENSON 6301 ;
- un traceur BENSON 1625-S.

Son développement a été effectué à partir de :

- QUICKBASIC (Basic compilé) ;
- FABS+ (Gestion des fichiers) ;
- GPR 200 T (Gestion du traceur) ;
- assembleur (Gestion de la mémoire).

Il tourne sous système d'exploitation MSDOS 2.1

2.2. Description de FORBAN

FORBAN se compose d'une trentaine de modules adressables à partir d'un menu principal et correspondant à chacune des étapes des différentes opérations à réaliser pour obtenir le produit.

2.2.1. Les sous-programmes communs

Un ensemble de sous-programmes a été développé pour être utilisable par tous les modules. Ces sous-programmes concernent entre autres :

- *les saisies* : définition du type, de la longueur, du positionnement, du format, etc. et retour de la réponse et du code fonction éventuel ;

- *l'affichage de menus d'aide*, qui peuvent être demandés à tout moment de saisie. Une page d'écran est réservée pour afficher le menu d'aide correspondant à l'option en cours d'utilisation ;

- *la gestion de la couleur* : à chaque type d'affichage (saisie, aide, message, erreur, nom de rubrique, etc.), correspond une représentation : couleur et fond des caractères, modifiables par l'utilisateur.

Cet ensemble de sous-programmes permet à FORBAN d'être homogène au niveau de son utilisation, et facilement maintenable.

2.22 Gestion des fichiers

FORBAN utilise FABS+. Il s'agit d'un utilitaire de gestion d'index de fichier, pouvant gérer plusieurs centaines de milliers de clés (méthode dite en arbre B). L'application nécessite :

- 21 fichiers de données par carte (Tableau 6), de type aléatoire (ou accès direct), auxquels sont associés :
 - 16 fichiers index, possédant plusieurs clés ;
 - 3 fichiers annexes destinés à d'autres sites informatiques (à structure séquentielle, et de type ASCII) ;
 - 1 fichier répertoire, recensant l'ensemble des cartes existantes.
- 3 fichiers utilitaires.

2.23 Gestion des périphériques

L'utilisation des périphériques (Digitaliseur et Traceur) est faite d'une manière minimum, dans un souci d'utilisation sur d'autres types de matériel :

- *sur digitaliseur* : réception des données, sans émission particulière, la gestion de la sortie communication étant à la charge du BASIC ;
- *sur traceur* : appel aux fonctions suivantes : positionnement, tracé de lignes, changement de plumes, tracé de chaînes de caractères, modification de la taille des caractères.

2.24 Gestion de la mémoire

L'utilisation de FORBAN nécessitant une mémoire centrale de moins de 192 Ko (pour le système), il a donc été possible de réserver 448 Ko de mémoire (organisée en 7 segments de 64 Ko). Cette mémoire est utilisée pour stocker et manipuler un certain nombre de données en ligne, évitant ainsi les accès au disque qui ralentissent considérablement les traitements.

2.25. Sous-programmes spécifiques

Plusieurs traitements, essentiels au fonctionnement de FORBAN, ont été développés spécifiquement. En particulier :

Recherche de zones incluses

Il s'agit du traitement le plus long de FORBAN. Il est utilisé à deux reprises, avant le calcul de surfaces des zones, sur carte de base et sur document thématique. Il permet de déduire la surface des zones incluses (pour les zones externes), et de permettre un hachurage correct des zones au niveau des tracés. Ce traitement est le plus long car il nécessite d'effectuer le contrôle d'une zone sur l'ensemble des autres zones.

FICHIERS	CONTENUS	UTILISATION
REPERTOIRE (cartes) POINTS ARCS NOEUDS ZONES / ARCS ZONES / IDENTIFIANTS ZONES / LIBELLES	code / intitulé / échelle coordonnées n° arc / points n° noeud / n° arc n° zone / n° arc n° zone / ident. n° zone / libellé / coord.	FICHIERS DE BASE
ZONES / SURFACES ZONES / POINTS ZONES / INCLUSIONS IDENT. / SURFACES	n° zone / points / surface coordonnées n° zone / incl. / principale n° identifiant / surface	FICHIERS APRES TRAITEMENT DE BASE
ATTRIBUTS / VALEURS IDENT. / ATT.-VAL. CLASSES	attribut / valeur / libellé n° ident. / att. / val. n° classe / att. / val.	FICHIERS INFORMATION THEMATIQUE DERIVEE
REQUETE ZONES / SURFACES ZONES / ARCS ZONES / POINTS ZONES / INCLUSIONS ZONES / CLASSES	n° classe / n° ident.-zone n° zone / points / surface n° zone / n° arc coordonnées n° zone / incl. / principale n° zone / n° classe	FICHIERS REORGANISES BASE DU DOCUMENT THEMATIQUE
DESSIN (Paramètres) CONTOUR Carte	classe/plume/distance/angle n° arc / sens de rotation	FICHIERS DE RESTITUTION

Tab. 6.- Liste des fichiers

Le principe consiste à déterminer un point interne à la zone, et de compter le nombre d'intersection entre la demi-droite partant de ce point, et la zone supposée externe. Si le nombre d'intersections est nul ou pair, la zone n'est pas incluse ; inversement, s'il est impair, la zone est incluse (Figure 2).

Calculs de surface

Ils permettent de déterminer la surface de chaque zone, la surface totale de la carte, la surface totale par identifiant ou par classe, les différents pourcentages. Une unité-terrain est choisie en fonction de l'échelle de base :

- m² jusqu'à 1/10 000.
- ha de 1/10 000 à 1/100 000.
- km² au dessus de 1/100 000.

Le calcul s'effectue de la manière suivante :

$$\begin{aligned} & i=N \\ \text{SOMME } [& X (i) * Y (i+1) - X (i+1) * Y (i)] / 2 \\ & i=1 \end{aligned}$$

où N est le nombre de points de la zone, X (i) et Y (i) sont les coordonnées du i^{ème} point de la zone.

Hachurage de zones

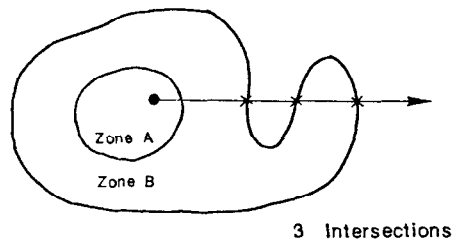
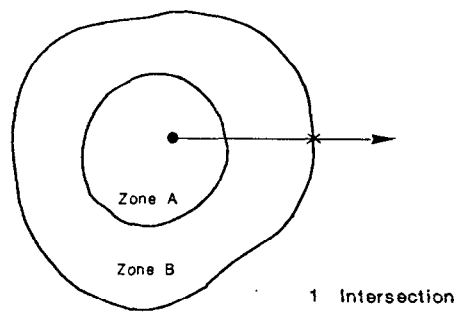
FORBAN utilise ce traitement sur l'ensemble des tracés. Le principe est proche de celui des recherches d'inclusions : recherche des intersections par rapport à une droite. Cette droite est déterminée en fonction des paramètres de dessin définis par l'utilisateur : angle d'inclinaison et distance entre hachures. Le principe est donc le suivant :

- positionnement de plume sur le point suivant (ou premier) ;
- tracé de ligne jusqu'à l'intersection suivante.

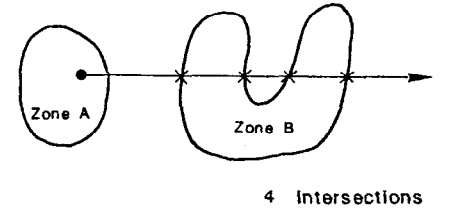
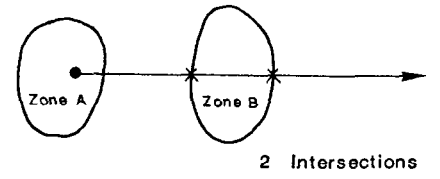
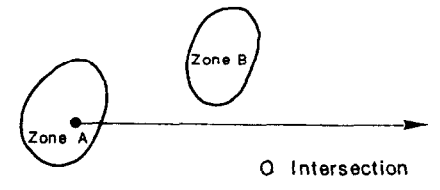
L'opération est répétée jusqu'à la dernière intersection trouvée.

Regroupement de zones :

Ce traitement sert à supprimer les arcs communs aux zones de même classe après la constitution des classe. Le principe est de partir d'un des arcs d'une zone considérée, et de se déplacer d'arcs en arcs en changeant de zone si la zone possédant le même arc est de même classe. L'arrêt du traitement s'effectue lorsque l'arc en cours correspond à l'arc de début, dont les zones communes sont de classes différentes (Figure 3).



Le nombre d'intersections est Impair
 La zone A est incluse dans la zone B

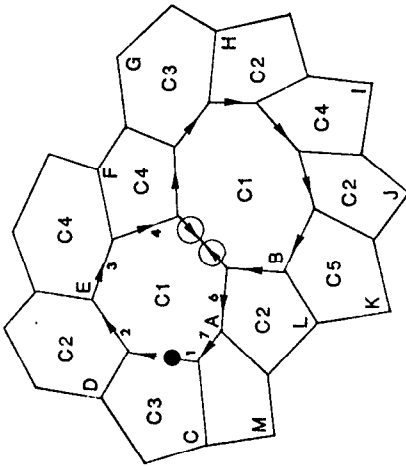


Le nombre d'intersections est Nul ou Pair
 La zone A n'est pas incluse dans la zone B

Fig. 2 - Principe de recherche des inclusions

- 1^{er} arc à considérer
- ⊙ Changement de direction
- ⊙ l'arc est commun à 2 zones de même classe

CN = Classe N
 N = Zone N
 1 = N°d'arc

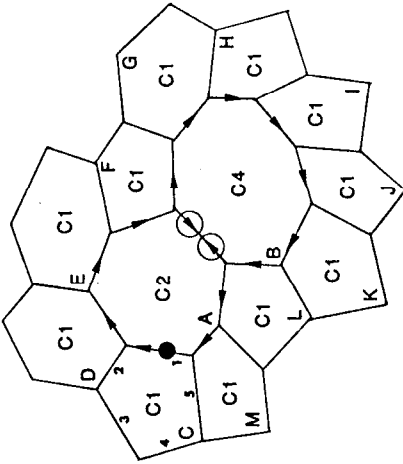


CAS N°1

Début de recherche sur ARC N°1 de zone A
 Enveloppe trouvée = Contour zones A+B

Zone A incluse dans Enveloppe trouvée

ENVELOPPE CONSERVEE



CAS N°2

Début de recherche sur ARC N°1 de zone C
 Enveloppe trouvée = Contour zones A+B

Zone C non incluse dans Enveloppe trouvée

ENVELOPPE REJETEE

Suite de recherche sur ARC N°2 de zone C

Fig. 3 - Principe de réorganisation des contours

Un contrôle est alors effectué pour vérifier que l'enveloppe de la nouvelle zone formée est bien l'enveloppe externe et non l'enveloppe interne dans le cas d'une zone en forme de couronne (donc possédant une inclusion). Ce contrôle est effectué par recherche d'un point interne de la première zone considérée, puis vérification que ce point est bien interne à la nouvelle zone (recherche du nombre d'intersection de la demi-droite à partir de ce point, impair).

2.3. Développements ultérieurs

Cette application, bien que développée pour une exploitation de cartes pédologiques, est d'ores et déjà utilisable pour tout traitement thématique de documents de base d'autres domaines (Sciences de la Terre, Végétation, etc.).

Elle reste ouverte à d'autres extensions dont le développement est prévu :

- *croisement et assemblage* de plusieurs cartes contiguës pour réalisation de documents de synthèse tant sur le plan de la thématique de base, que de thèmes dérivés, en utilisant les possibilités déjà effectives de réduction d'échelle ;

- *remaniement et transfert des fichiers* des divers documents saisis et créés à des systèmes de cartographie assistée par ordinateur pour gravure extérieure de typons en vue d'édition imprimée ;

- *transformation de systèmes de projection* en vue de comparaison et superposition de divers documents cartographiques établis sous ces différents systèmes : recalage sur fond topographique après changement d'échelle, etc..

Ces développements permettront d'ouvrir aux produits de FORBAN des débouchés sur des systèmes et utilisateurs extérieurs.

3. CARACTERISTIQUES D'EXPLOITATION DE FORBAN

L'application a été développée, du fait du matériel disponible, sur un micro-ordinateur de type XT, sous système d'exploitation MS DOS et langage BASIC, qui ont entraîné leurs contraintes spécifiques : capacité mémoire, taille des fichiers et vitesse de traitement limitées. Des solutions originales y ont porté remède : utilisation d'un Basic et d'un compilateur plus performants (Quick-Basic), utilisation de segments mémoires, limitation du volume des données, réduction du nombre d'accès aux fichiers.

D'autres améliorations pourront être apportées. L'acquisition d'une unité centrale de type AT, pour laquelle FORBAN est d'ores et déjà compatible, diminuera les temps de traitement dans un rapport allant de 3 à 5. L'adjonction d'une unité externe de traitement et de stockage à forte capacité (modules de 30 Mo) permettra l'archivage de la carte de base et d'un nombre de documents thématiques dérivés plus important. La mise en place de cartes mémoire supplémentaires permettra le transfert en mémoire centrale des fichiers en cours de traitement en évitant complètement les accès au disque et en accélérant ainsi considérablement les temps de traitement.

Il en va de même en ce qui concerne le déroulement des opérations, saisies, traitements et restitutions, qui ne peuvent actuellement pas être effectuées simultanément. Elles le pourront être aisément en présence de deux postes de travail (l'actuel XT dédié aux fonctions de saisie et un AT, plus performant, dédié aux différents traitements et restitutions). Le caractère entièrement modulaire de FORBAN permet la séparation de ces diverses tâches.

CONCLUSION

La mise au point de systèmes de traitement d'information géographiquement localisée, tel le logiciel FORBAN, constitue un grand progrès dans l'élaboration de documents cartographiques thématiques, en particulier en matière de Pédologie, où ce type de carte est très fréquemment demandé en complément de la carte pédologique de base. Ce procédé infographique permet le traitement complet et automatique de l'information saisie, tant sur le plan géographique (tracé de contours), que thématique (extraction, croisement, regroupement, combinaison, superposition de thèmes). Il procure ainsi un gain de temps et une flexibilité considérables qui permettent de répondre au cas par cas à la demande des utilisateurs, le document de base étant introduit en mémoire d'accès permanent, tout en déchargeant les Services de Cartographie traditionnels de la réalisation de ces cartes thématiques.

L'exploitation de FORBAN ne nécessite qu'un matériel informatique de large diffusion, et fait donc preuve d'une grande compatibilité. Développé dans le cadre de l'ORSTOM par les auteurs, il met en oeuvre un ensemble d'utilitaires courants et de logiciels spécifiquement développés pour son application.

FORBAN constitue donc un outil de cartographie assisté par ordinateur complet et performant. Il se caractérise enfin par une remarquable simplicité d'utilisation du fait de la particulière clarté de ses commandes.

SISECSO OU L'INFORMATIQUE AU SERVICE D'UNE RECHERCHE EN GÉOGRAPHIE HUMAINE : LA DYNAMIQUE SPATIALE DES CERRADOS AU BRÉSIL

Philippe WANIEZ

(Géographie)

(UR 502, Cadres spatiaux de l'Indépendance Alimentaire)

avec la collaboration de Jozeneida PIMENTA de AGUIAR
technicienne à l'EMBRAPA/CPAC, Planaltina (D.F.), Brésil
et de Violette CABOS
cartographe, ingénieur au CNRS, Montpellier, France

RESUME - Un système d'information géographique : pour quoi faire ?

SISECSO est le module "*socio-économique*" d'un ensemble plus vaste nommé SISGEO, réalisé conjointement par l'ORSTOM et l'Empresa Brasileira de Pesquisa Agropecuaria (EMBRAPA) et visant la mise au point d'un système d'information sur l'environnement. Il s'agit de "*capitaliser*" toutes les informations existantes relatives aux conditions de développement de l'agriculture brésilienne, conditions physiques, mais aussi humaines.

Initialement, le projet SISECSO apparaissait plus comme une réponse à une demande de type technocratique qu'à un programme de recherche centré sur une problématique. Rien de bon ni d'intéressant n'aurait pu sortir d'une telle approche faisant l'impasse sur 1/ *la définition précise d'un problème scientifique* et l'énoncé de quelques hypothèses y afférent, 2/ *élaboration d'un corpus statistique* en parfaite adéquation avec les concepts qu'il est censé représenter, 3/ *l'immédiate mobilisation* de la totalité de l'arsenal *des techniques d'analyse*.

Ce diagnostic a eu pour conséquences le choix d'autres finalités et d'autres moyens que ceux prévus initialement.

Problématique et méthodologie : la dynamique régionale des cerrados

Les *cerrados* sont composés de la partie périphérique de plusieurs régions brésiliennes très diverses couvrant une grande variété de niveaux de développement. Le géographe est directement conduit à observer et comprendre le processus de formation de ce qui n'est pas une vraie région géographique mais qui pourrait le devenir.

La base de données réalisée dans le cadre de SISECSO constitue à la fois le moyen et un produit de la recherche. C'est le moyen car il est préférable d'asseoir l'essentiel du travail de recherche sur la petite échelle. C'est aussi un produit de la recherche car elle renferme l'essentiel des informations nécessaires à la compréhension du processus d'émergence régionale.

Cette approche doit être mise en perspective avec ce que certains ont pu appeler la "*révolution quantitative*" affectant la géographie française depuis une quinzaine d'années.

Sources statistiques, aire d'étude et unités d'observations

L'Institut Brésilien de Géographie et de Statistique (IBGE) est l'organe central de gestion et de restitution des données générées par le fonctionnement de l'appareil administratif. L'IBGE assure également la réalisation périodique de recensements.

Ainsi, l'IBGE conserve les fichiers et les diffuse auprès des entreprises qui en font la demande. Dans le cadre de SISECSO, toutes les données censitaires de 1970 à 1980 ont été acquises : agriculture, industrie, services et démographie. Elles ont été complétées par les données relatives aux productions de l'agriculture et de l'élevage.

La définition précise de l'aire d'étude s'est appuyée sur les travaux antérieurs du Centre de Recherches Agro-pastorales des cerrados (CPAC). C'est un ensemble de 885 *municípios*, unité politico-administrative de base, formant une maille extrêmement évolutive, caractéristique dont découlent quelques problèmes délicats pour l'organisation des données.

Adopter un modèle d'informatisation.

Préalablement au choix d'un modèle de mise en oeuvre de l'informatique, il a fallu procéder à une analyse des diverses possibilités. L'option "*progiciel clés en mains*" a été choisie en raison de la brièveté des délais nécessaires à sa mise en oeuvre.

Le Statistical Analysis System (SAS) utilisé depuis longtemps à l'EMBRAPA et correspondant bien à la nature des données à organiser constitue l'outil informatique de réalisation de SISECSO, implanté sur un gros système IBM (2 ordinateurs IBM 4341) et accessible en batch sous OS/VS2 et en conversationnel sous VM/CMS.

La place de SAS au cours des principales étapes de réalisation de SISECSO

En raison de la grande variété de fonctions proposées par SAS, toutes les étapes de réalisation de SISECSO y ont fait appel, à des degrés divers : 1/ *la constitution du tableau d'identification des municípios*, 2/ *la lecture des bandes de l'IBGE*, 3/ *la constitution des tableaux finals*.

Seule la numérisation de la carte des municipios des cerrados a nécessité la conception et la programmation d'un petit ensemble de programmes générant un fichier "*fond de carte*" compatible avec SAS/GRAPH.

Composition et avenir de SISECSO

Au printemps 1987, SISECSO se compose de 3 bases de données informatisées, implantées sur le centre informatique de l'EMBRAPA et accessibles via le progiciel SAS. A la fin de l'année 1987, le CPAC disposera d'une banque de données de près de 3 000 variables relevées sur 856 municipios. Mais ces données relatives à la décennie 1970-1980 devront être complétées au fur et à mesure de la sortie des nouveaux recensements.

1. UN SYSTEME D'INFORMATION GÉOGRAPHIQUE : POUR QUOI FAIRE ?

SISECSO est un programme de coopération initié il y a plusieurs années, visant "*la mise au point d'un système d'information sur l'environnement*". La finalité de ce projet apparaît parfois comme étant la "*réalisation d'un système d'informations géographiques à vocation agricole*". La prétention de SISGEO est donc de couvrir à la fois l'ensemble des questions relatives à l'environnement de l'activité agricole, ainsi que la totalité du territoire brésilien. La division du travail est conçue de la manière suivante : d'une part, six modules thématiques traitent du climat, de la végétation, des sols, de l'hydrologie, des données phyto-sanitaires et de l'information socio-économique ; d'autre part, deux modules méthodologiques sont voués l'un à la cartographie automatique, l'autre à la télédétection. Avec sept départements, services ou centres de recherche de l'EMBRAPA, 56 chercheurs, ingénieurs ou techniciens brésiliens et quatre de l'ORSTOM, SISGEO s'inscrit naturellement dans le cadre de l'Unité de Recherche 502 de l'ORSTOM (Cadres spatiaux de l'indépendance alimentaire).

Le module relatif aux données économiques et sociales, SISECSO, a été entrepris bien après ceux ayant trait aux sols et au climat ; c'est au Centre de Recherche Agro-Pastorale des Cerrados (CPAC) qu'est revenu sa conception. En effet, la forte expansion de l'agriculture des cerrados en fait un excellent objet de recherche, cristallisant l'essentiel des problèmes scientifiques que peut présenter un tel projet. Malheureusement, une fois à pied d'oeuvre, on s'est rapidement rendu compte que dans cette institution, la "*socio-économie*" n'était qu'un tiroir de plus, un nouveau facteur à optimiser pour accroître les rendements, améliorer les conditions de la commercialisation, mieux définir la demande des consommateurs

potentiels. On avait donc affaire à une demande purement technocratique pouvant se résumer de la manière suivante : accumulons la plus grande masse d'information dans tous les domaines ayant un rapport avec l'activité agricole ; interrogeons la banque de données ainsi constituée et produisons des "*caractérisations*" plus ou moins fouillées en fonction de la "*demande sociale*" (en fait la demande de l'état planificateur) ; puis, effectuons des "*croisements synthétiques*" montrant sur quels éléments il faudrait agir pour atteindre un optimum donné. Dans cette perspective, l'informatique n'intervient que pour rendre plus efficiente, plus systématique l'oeuvre de planification.

Sur un tel programme, plusieurs points de vue peuvent être adoptés. Celui de l'archiviste : plus il y a de données, meilleur c'est ; cela servira peut-être un jour et tant mieux si l'informatique permet de s'y retrouver plus facilement. Celui du technocrate : avec une telle masse de données, je vais pouvoir truffier mes rapports de tableaux de chiffres, judicieusement choisis pour venir à l'appui de mes conceptions. Celui du technicien agricole : enfin, je vais pouvoir définir avec précision où je peux planter mes carottes. Celui du responsable scientifique : l'avenir c'est l'informatique ! d'ailleurs, voyez la recherche s'informatise à toute allure. Celui du chercheur agronome ou sociologue classique : tout cela ne remplacera jamais ma solide expérience de terrain ; ce qu'il faut, c'est "*sentir les liaisons*" et les illustrer par quelques données bien choisies. Curieusement, le géographe concevant sa discipline comme l'analyse des structures de l'espace sera peut-être le plus déçu par ce genre de réalisation. En effet, pour conduire à quelques découvertes intéressantes, la méthodologie quantitative requiert au moins trois conditions préalables : 1/ la définition précise d'un problème scientifique et l'énoncé de quelques hypothèses y afférent, 2/ la constitution d'un corpus statistique en parfaite adéquation avec les concepts qu'il est censé représenter, 3/ l'immédiate mobilisation de la totalité de l'arsenal des techniques d'analyse. Dans le cas du programme SISECSO, aucun de ces trois points n'était pleinement représenté ; le premier était totalement ignoré, le second ne faisait l'objet que de discussions de détail sur la véracité de telle ou telle valeur relevée dans un annuaire statistique, et le troisième était réduit à sa plus simple expression : statistique descriptive et tabulations.

En appliquant à SISECSO les propos du sociologue Wright Mills, on peut se demander si ses promoteurs n'étaient pas victimes de "*l'inhibition méthodologique*", c'est-à-dire d'une confusion entre l'objet et la méthode. Le fait que le projet insiste plus sur l'acquisition et l'organisation informatique des données que sur l'usage qui doit en être fait dans le cas d'une recherche précise montre bien où se situe le problème. Si l'on ajoute à cela la position

dominante des informaticiens et la multiplication des modules spécialisés segmentant le poids des divers interlocuteurs, on aboutit à une situation où le chercheur est littéralement dépossédé de son objet de recherche qui devient la matière première d'une équipe de techniciens naturellement plus enclins à mettre l'accent sur la méthode que sur l'objet. L'épuisement de ce processus se traduit par l'existence d'un produit informatique peu utile à la recherche, mais que peuvent s'accaparer les technostructures pour leurs fins propres. C'est pourquoi il ne faut pas s'étonner des encouragements réitérés exprimés par les structures bureaucratiques de la recherche, de l'EMBRAPA notamment. Si l'on admet sans grande discussion le bien fondé de telles réalisations à des fins de gestion du territoire et, dans une moindre mesure, de planification, on doit fermement contester leur utilité pour la recherche en géographie humaine. Il y a là une confusion des genres : la recherche, base du développement, devrait être autre chose qu'une activité de service informatique pour le seul profit de l'Etat planificateur. Ce diagnostic a eu pour conséquence directe le choix d'autres finalités et d'autres moyens pour la réalisation de SISECSO.

2. PROBLÉMATIQUE ET MÉTHODOLOGIE : DYNAMIQUE RÉGIONALE DES CERRADOS

Etudier, décrire, même de manière très fouillée, les transformations des cerrados, ne constitue en aucun cas une problématique de recherche en géographie humaine. Après avoir pris connaissance de la maigre littérature sur le sujet, et demandé le point de vue de plusieurs chercheurs et aménageurs (français...), il a été possible de définir un ensemble de questions méritant d'être étudiées.

Les cerrados constituent un immense ensemble biogéographique s'étendant sur deux millions de kilomètres carrés, près d'un quart du territoire brésilien. Sur le plan physiognomique, le cerrado typique est une savane composée d'une couverture herbacée de 50 à 70 centimètres de hauteur, et d'un étage supérieur discontinu d'éléments arborés ou arbustifs. Ce "*cerrado tipo*" donne à l'ensemble son unité, et résulte, sur le plan écologique, de la conjonction de sols acides et d'un climat à deux saisons, l'une sèche, l'autre humide ; ces conditions sont réunies sur un peu plus des deux tiers du territoire. Ailleurs, des variantes générées par des facteurs limitants comme la qualité des sols ou les conditions hydrologiques sont à l'origine des formations plus ouvertes telles que les "*campo cerrado*", "*campo sujo*", "*campo limpo*", ou plus fermées comme le "*cerradão*" sur à peu près un quart de la superficie. L'aire d'extension des cerrados est une bande de largeur variable, d'orientation NE/SW, couvrant la majeure partie des états du

Goiás, Mato Grosso, Mato Grosso do Sul, Minas Gerais, et une partie importante du *Maranhão, Piauí, Bahia*. Cette rapide description de l'homogénéité biogéographique relative n'aurait pas d'intérêt si elle n'avait pas été à l'origine de la transformation de ce grand espace en un enjeu considérable pour l'Etat brésilien : le contrôle et l'intégration nationale de "*l'Interior*".

Limités à leur partie la plus massive, les cerrados appartiennent à plusieurs grandes régions définies par l'Instituto Brasileiro de Geografia e Estatística (IBGE), (Figure 1), à la fois au Nordeste, au Sudeste et au Cento-Oeste. Cette observation conduit directement à se poser la question de l'intérêt (autre que technocratique) de réaliser une banque de données socio-économiques à propos d'un espace chevauchant les principales divisions régionales du Brésil. C'est précisément à partir de cette remarque qu'est née la problématique de recherche : les cerrados sont composés de la partie périphérique de plusieurs régions brésiliennes très diverses, couvrant une grande variété de niveaux de développement. Qu'y a-t-il de commun entre, d'une part, la frange sud du Goiás ou du Mato Grosso do Sul, jouxtant l'Etat de São Paulo, phare de l'économie brésilienne, et, d'autre part, les zones dépressives du Piauí ou de l'ouest de Bahia ? Rien, si ce n'est d'appartenir à une même région biogéographique, les cerrados. Cette caractéristique est si forte, que nombreux sont ceux qui, au niveau des grands organismes d'aménagement de l'Etat, n'hésitent pas à parler de la "*Région de Cerrados*". S'agit-il d'un simple abus de langage, ou bien, comme c'est peut-être le cas, de la volonté déclarée d'ériger les cerrados en témoin avancé d'une rupture avec le sous-développement, un peu comme l'avait déjà fait le Président Juscelino Kubitschek, en installant la nouvelle capitale, Brasília, au coeur des cerrados ? Si cette hypothèse s'avérait exacte, les cerrados seraient une sorte de creuset, ou mieux, un laboratoire vivant, où l'on pourrait observer des transformations rapides résultant des jeux d'influences convergentes et se répercutant sur tous les aspects de la vie économique et sociale.

Le géographe est directement conduit à observer et comprendre le processus de formation de ce qui n'est pas encore vraiment une région mais qui pourrait le devenir. L'inscription dans l'espace géographique des effets des forces d'intégration en présence est une problématique de recherche dont les résultats peuvent apporter des informations utiles au développement des cerrados.

Une problématique sans méthodologie, c'est poser une série de questions sans se donner la possibilité de leur apporter une réponse. La banque de données SISECSO constitue à la fois le moyen et le produit de la recherche. C'est le moyen car il apparaît préférable d'asseoir l'essentiel du travail de recherche sur la petite



échelle, c'est-à-dire d'abord sur la totalité des deux millions de kilomètres carrés. Ce n'est pas une pratique habituelle à l'ORSTOM, mais c'est la meilleure façon d'apprécier la formation de la région prise dans son ensemble. SISECSO sera aussi le produit de la recherche en renfermant la plupart des informations nécessaires à l'analyse et à la compréhension des processus.

Cette approche différente doit être mise en perspective avec ce que certains ont pu appeler la *"révolution quantitative"* affectant la géographie française depuis une quinzaine d'années. Il s'agit d'une profonde mutation, qui ne peut être réduite à un recours au nombre plus fréquent qu'auparavant. En modifiant leur approche des phénomènes, en adoptant le modèle d'analyse systémique, en mettant l'accent sur la nécessité de ne pas privilégier un niveau d'analyse ou une échelle particuliers aux dépens des autres, c'est une autre manière de penser et de pratiquer la géographie qui a fait son apparition. La lecture de la revue *"L'Espace Géographique"* donne une bonne idée des travaux produits par cette *"nouvelle géographie"* durant la décennie 1970- 1980. La différence la plus aisée à apprécier relève de la présentation de tableaux numériques ou de graphiques comme on en avait peu vu jusqu'alors, issus principalement des techniques d'analyse statistique multivariée : droites de régression, plans factoriels, arbres de classification. C'est probablement pour cette raison que la discipline scientifique *"géographie"* a été affublée de l'épithète *"quantitative"* par ses détracteurs, tenants de la géographie *"classique"* qui doit se pratiquer *"les pieds dans la boue !"* Les pionniers de la géographie quantitative française ayant démontré l'intérêt et l'efficacité de leur méthode et de leurs techniques d'analyse, toutes les branches de la discipline y font aujourd'hui appel, mais à des degrés assez divers. SISECSO est un exemple de pénétration de la géographie quantitative dans un secteur jusqu'alors épargné !

La méthodologie peut être résumée de la manière suivante : à partir d'informations (statistiques pour l'essentiel) relevées sur la totalité du territoire des cerrados, à des dates différentes permettant d'évaluer les évolutions, et enregistrées dans une banque de données informatisée, on cherche à mettre en évidence les principaux mouvements d'intégration régionale, à l'aide de techniques mathématiques et graphiques d'analyse de l'espace. Cependant, rien n'interdit la vérification de certains résultats en allant a posteriori sur le terrain, mais sûrement pas l'inverse. On pourrait objecter que les données statistiques brésiliennes ne constitueraient pas une source d'information fiable. Il semble plus constructif d'examiner s'il n'est pas possible de leur faire dire quelque chose. Après tout, n'est-ce pas une des formes de la pratique de la recherche : analyser en connaissance de cause des données partielles, entachées d'erreurs, à l'aide de techniques appropriées ?

3. SOURCES STATISTIQUES, AIRE D'ÉTUDE ET UNITÉS D'OBSERVATION

Le Brésil est un pays de tradition statistique ancienne. La philosophie positiviste d'Auguste Comte, encore si présente aujourd'hui, n'est peut-être pas étrangère à cette demande de données numériques. En 1808, un dénombrement montrait l'existence de 4 millions d'habitants, mais il ne s'agissait que d'une estimation visant à informer la métropole portugaise de l'état de ses domaines en Amérique du Sud. Ce n'est qu'en 1872 qu'eut lieu le premier vrai recensement démographique, digne de ce nom en raison du contrôle effectif de chacune des phases de sa réalisation. Suivirent, en 1890 et 1900, deux autres recensements de la population, puis en 1920 furent ajoutés ceux de l'agriculture et de l'industrie. 1940 marque un tournant de l'appareil statistique brésilien : outre le recensement démographique, furent réalisés les recensements économique, agricole, industriel, commercial, des transports et des communications. Cette immense tâche avait été rendue possible par la fondation, en 1934, de l'Institut National de Statistique, devenu, en 1938, l'Institut Brésilien de Géographie et de Statistique (IBGE). Chargé à la fois des enquêtes et études statistiques, géographiques et cartographiques, l'Institut a conservé son nom jusqu'à nos jours, en étendant ses activités au domaine de la planification. En 1950, 1960 et 1970, l'IBGE réitéra ses opérations de 1940 en les affinant et les adaptant aux nouvelles caractéristiques économiques et sociales du pays. En 1975, les recensements économiques devinrent quinquennaux. En 1980 eut lieu le dernier en date des recensements démographiques, en même temps que les recensements économiques, dont les plus récents datent de 1985 (et ne sont pas encore disponibles).

Réduire la production statistique de l'IBGE à la réalisation des recensements serait une grossière erreur. Comme tous les grands instituts du même type dans le monde, l'IBGE tend à devenir l'organe central de gestion et de restitution des données générées par le fonctionnement de l'appareil administratif. Cette évolution prend en particulier la forme de la coordination des services statistiques existant dans chaque état de la Fédération, mais également de la conservation de divers fichiers dans son grand centre informatique de Rio de Janeiro. Par exemple, l'IBGE publie, depuis 1973, deux recueils relatifs aux productions municipales agricoles et à l'élevage (PMA et PMP), réalisés à partir du croisement d'informations multiples, en provenance des agriculteurs, des conseillers agricoles et des coopératives. Ces données, paraissant avec deux ans de retard, sont considérées par les spécialistes de

l'économie agricole comme étant plus fiables que les recensements de l'agriculture.

La part la plus importante des statistiques de l'IBGE fait apparaître des tableaux au niveau des états de la Fédération ; les agrégats nationaux n'ont de sens que s'ils sont examinés à la lumière des disparités territoriales. Heureusement, pour tout ce qui concerne les données de base (démographie et études économiques notamment), l'IBGE utilise comme unité territoriale le "*município*" : c'est l'unité politico-administrative la plus petite, la moins hétérogène (bien que très disparate dans les états les moins peuplés), celle qui permet d'asseoir une recherche géographique.

En l'absence de travaux préliminaires au CPAC, c'est à ces données de base disponibles au niveau municipal qu'on s'est attaché en premier lieu.

La délimitation précise de l'aire d'extension des cerrados est difficile et a été à l'origine de nombreuses controverses. D'un côté, on trouve ceux qui réduisent les cerrados à leurs formes les plus typiques, excluant les marges et les enclaves ayant d'autres compositions biogéographiques. A l'opposé, on peut penser qu'une délimitation large favorisera ultérieurement le choix d'un périmètre plus spécifique, basé non plus sur les seules caractéristiques physiques homogènes, mais aussi sur l'appartenance à une région géographique. Cette seconde option paraissant plus opératoire pour la recherche envisagée et correspondant aux choix du CPAC, il a suffi de calquer les limites des cerrados définies par l'EMBRAPA sur une carte municipale du Brésil (à l'échelle du 1/2 500 000) obtenue à l'IBGE après bien des péripéties (la plupart des travaux faits par les Brésiliens se limitent le plus souvent aux états ou groupes d'états formant des régions ; demander une carte à une échelle unique, couvrant à la fois les grands espaces du Mato Grosso et ceux, plus densément peuplés du Minas Gerais, apparaissait un peu extravagant...mais cette carte existait au service des atlas de l'IBGE, perdu dans la périphérie nord de Rio de Janeiro).

Contrairement au maillage ancien et peu évolutif de la France (le Brésil n'ayant pas encore achevé son processus de contrôle territorial), la maille municipale des cerrados évolue assez rapidement. Sur les 885 *municípios* référencés dans le recensement de 1980, plus de 400 n'existaient pas en 1950 ; de 1970 à 1980, plus d'une quarantaine de *municípios* ont été créés. Il n'est pas possible d'ignorer ces transformations car elles faussent les éventuelles comparaisons entre les recensements de 1970 et 1980. De plus, les *municípios* concernés sont précisément ceux où l'expansion démographique a eu pour conséquence la reconnaissance d'une communauté, se traduisant par la création d'une municipalité. C'est la raison pour laquelle il a fallu recalculer les valeurs données par le recensement de 1980 dans le cadre de la maille municipale de 1970.

Lorsque les nouveaux étaient de simples partitions d'un seul municipio ancien, la solution était simple : il suffisait d'additionner les chiffres de 1980. Par contre, lorsqu'un nouveau municipio était constitué avec des parties de plusieurs autres (comme c'est souvent le cas dans le Mato Grosso), il a fallu modifier la maille de 1970 pour former un municipio théorique comparable avec la réunion de ceux ayant donné une portion au nouveau. Par exemple, dans l'état du Mato Grosso, les municipios *Nobres et Chapada do Guimaraes* ont vu leur superficie se réduire comme une peau de chagrin entre 1970 et 1980 ; quatre nouveaux municipios ont été créés à leurs dépens : *Paramatinga, Nova Brazilandia, Sinop et Colider*, Sinop chevauchant d'ailleurs le territoire des deux municipios originaux. Pour comparer les données de 1970 et de 1980, un municipio théorique rassemble les deux de 1970 et les six de 1980 (fig. 2). De 885 municipios en 1980, la banque de données n'en connaît donc que 856 strictement identiques en 1970 et 1980. Bien entendu, le même travail devra être réalisé pour rendre semblables les mailles de 1980 et 1985 ou 1970 et 1985.

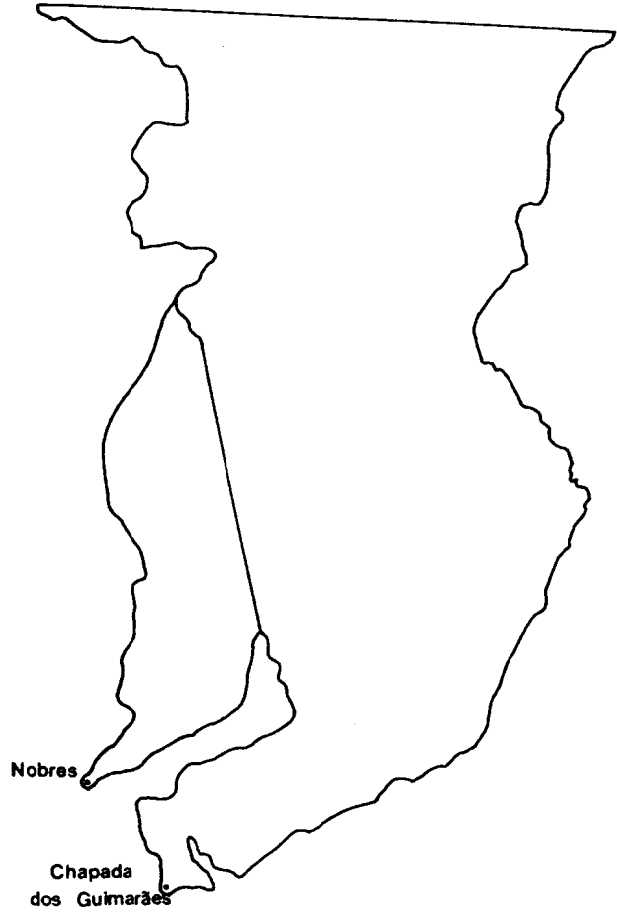
4. ADOPTER UN MODELE D'INFORMATISATION

Préalablement au choix d'un modèle de mise en oeuvre de l'informatique, il a fallu procéder à une analyse des diverses possibilités. Résultat : trois options différant sur le plan des délais nécessaires à la programmation, du niveau technique requis et du mode d'utilisation des données.

La première option, connue sous le nom de "*système d'information*", a pour vocation d'enregistrer la totalité des documents de nature diverse traitant d'un thème général, par exemple l'environnement, ou d'un espace tel qu'une ville ou une région. Un cas particulier est celui des systèmes d'information géographique, dans lesquels sont rassemblées à la fois des cartes, des enquêtes de terrain, des séries statistiques, des mesures de télédétection, etc.. De telles réalisations, devant répondre non seulement aux besoins des chercheurs mais aussi à ceux des administrateurs et même des citoyens, requièrent des moyens humains et matériels considérables, ainsi qu'une politique à long terme ; en effet, l'ensemble ne peut être vraiment opérationnel, c'est-à-dire apte à proposer un large éventail d'applications, qu'après de longues étapes de mise au point du progiciel gouvernant le stockage et la restitution de l'information, mais aussi l'enregistrement des documents de base à proprement parler.

La seconde option, diamétralement opposée à celle qui précède, met l'accent sur la capacité de l'ordinateur à réaliser un traitement plus rapide de données plus volumineuses, rassemblées

1970



1980

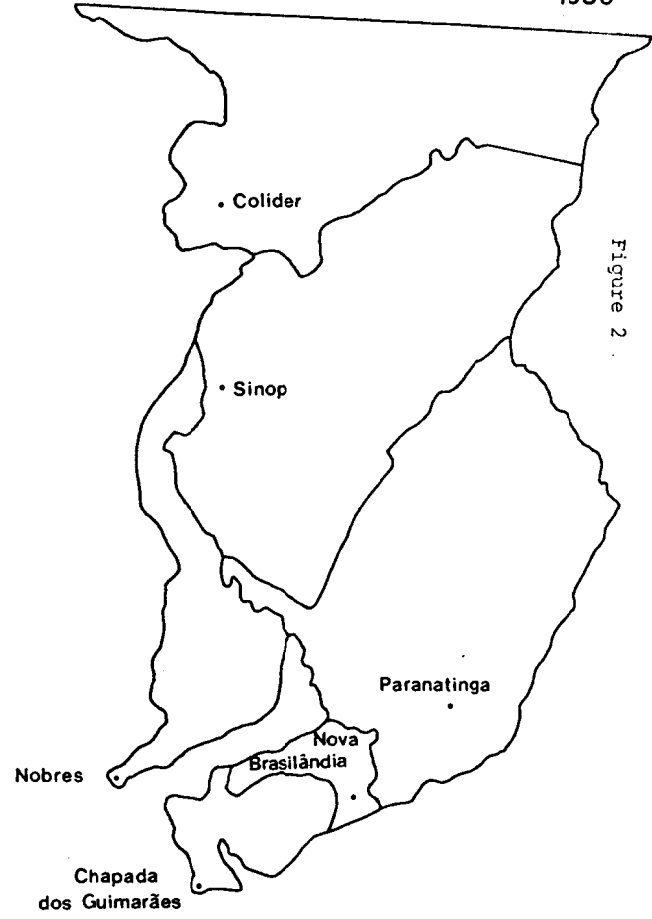
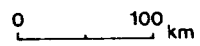


Figure 2



dans le cadre spécifique d'une problématique de recherche. La fonction d'archivage est, dans ce cas, réduite au minimum, c'est-à-dire à l'élaboration d'un fichier enregistré sur un support magnétique où les programmes de traitement, le plus souvent statistiques, viendront lire les données. Ce procédé a montré ses avantages mais aussi ses limites : les résultats sortent plus vite, mais les données ayant permis de les obtenir, souvent collectées au prix d'un grand effort, sont délaissées par la suite et deviennent inaccessibles à la communauté scientifique, faute du minimum de gestion requis : l'analyse secondaire est impossible.

Enfin, une troisième option s'impose depuis quelques années, celle des bases de données alliant la rapidité des réalisations à la rigueur de gestion. Des progiciels "*clés en main*" permettent d'enregistrer facilement une information rassemblée dans le cadre d'un problème particulier, tout en facilitant sa préparation pour l'analyse ; certains d'entre-eux intègrent en un seul ensemble les fonctions d'archivage et d'analyse. Ainsi est-il possible d'obtenir dans un délai raisonnable, de l'ordre de 18 mois à deux ans, des résultats de recherche, et cela tout en préservant l'avenir. En effet, rien ne s'oppose à la récupération des données par un système d'information plus ambitieux, avec un avantage substantiel : on connaît mieux leur signification et leur domaine de validité. Dans le cadre institutionnel de SISECSO, cette dernière option présentait de nombreux avantages sur les autres et a été finalement adoptée, provoquant une rupture avec l'équipe d'informaticiens de SISGEO, favorables à la première possibilité.

Acheter les droits d'utilisation d'un progiciel, c'est s'assurer, de manière contractuelle, un certain nombre d'avantages. Par exemple, on est sûr que les programmes ont été testés par des professionnels, c'est-à-dire que la probabilité de l'occurrence d'une erreur ou d'un incident est très réduite. En second lieu, la documentation livrée avec le système est souvent abondante, parfois didactique, et lorsqu'elle ne l'est pas, parfois relayée par des ouvrages en librairie. C'est ensuite la faculté d'échanger ses expériences avec d'autres utilisateurs et de progresser ainsi dans la maîtrise de cet outil. Enfin, et toujours dans le même ordre d'idée, il est quelquefois possible de recourir aux services d'une assistance technique qui peut à la fois dépanner les usagers et repertorier leurs critiques. Ces avantages évidents ne dispensent pas pour autant de l'élaboration d'un cahier des charges pour le choix définitif du produit ou, plus exactement de l'exposé des points clé, ceux qui conduiront à une discrimination immédiate de l'offre du marché. Bien entendu, cet examen est directement dépendant du type d'information à traiter, elle-même fonction de la problématique du chercheur, les données constituant un matériel adapté à une recherche et non l'inverse. Par exemple, si on considère le cas des

matrices d'information spatiale, tableaux rectangulaires où les lignes figurent des unités spatiales et les colonnes des attributs géographiques mesurés sur ces unités, les points clé sont les suivants : 1/ *facilité de gestion des matrices*, de sélection, d'association et de codification des attributs ou des unités, 2/ *grande variété de traitements statistiques*, univariés ou multivariés, descriptifs ou inférentiels, 3/ *représentation graphique des résultats* sous la forme de cartes en deux dimensions (cartes ponctuelles ou choroplèthes) ou trois (blocs diagrammes), 4/ *interactivité* (langage établissant le dialogue entre le système et le chercheur), *convivialité* (facilité du dialogue), *ouverture et portabilité* complétant cette liste non limitative. Il n'y a que très peu de produits sur le marché répondant à la totalité de ces spécifications.

Les avantages du *Statistical Analysis System* ont été décrits ("*SAS, un outil à la hauteur des besoins des géographes*") dans l'ouvrage de G. Dandoy intitulé "*l'Infographie à l'ORSTOM*".

Aux arguments généraux présentés dans ce texte, il faut en ajouter un plus particulier à SISECSO : le volume relativement important d'informations a fait préférer le Centre de Calcul de l'EMBRAPA (CCE) de Brasilia à une solution bâtie autour d'un micro-ordinateur. Doté de deux ordinateurs IBM 4341, l'un destiné au temps partagé sous VM/CMS, l'autre au batch sous OS/VS2, le CCE propose SAS à ses utilisateurs qui le pratiquent depuis de nombreuses années (malheureusement limités aux produits BASE, FSP et ETS). SAS apparaît ainsi comme un langage commun à un grand nombre de chercheurs de l'EMBRAPA pouvant exploiter eux-mêmes des bases de données SAS, en toute indépendance vis à vis des informaticiens.

5. PLACE DE SAS DANS LES PRINCIPALES ÉTAPES DE SISECSO

En raison de la grande variété des fonctions proposées par SAS, toutes les étapes de réalisation de SISECSO y ont fait appel, à des degrés divers.

5.1. La constitution du tableau d'identification des municipios

L'identification des unités d'observation est un préalable à toute constitution de banque de données. A l'aide de PROC FSEDIT, ce tableau est saisi en "*plein écran*", de la manière suivante :

1/ *le nom du municipio* sur 40 caractères,

2/ *son code* sur la carte de repérage, composé de l'identifiant de l'État de la Fédération sur deux lettres, suivi du numéro d'ordre sur sept chiffres,

3/ *ses codes numériques* figurant dans les fichiers de l'IBGE en 1980 et 1970,

4/ *d'autres informations complémentaires* comme les coordonnées géographiques du siège et la population en 1980.

FSEEDIT est très bien adapté à ce travail nécessitant un temps de saisie assez long et de nombreuses corrections successives, en particulier de l'orthographe des noms des municipios. Les nouveautés apportées par la version 5, notamment l'initialisation du tableau à l'appel de la procédure et surtout l'option AUTOSAVE ajoutent un confort certain à ce travail assez ingrat.

5.2. La lecture des bandes des recensements de l'IBGE

Les données fournies sur bande magnétique par l'IBGE sont enregistrées de deux manières. S'il s'agit de données récentes diffusées à l'aide du progiciel local SIDRA, on a affaire à un format "*caractères*". Pour les informations plus anciennes (de 1970 et 1975), l'enregistrement, réalisé par des programmes PL/1, se présente en format binaire. Les formats internes à SAS, ici \$8., IB4. ou RB4., simplifient considérablement l'opération de lecture des bandes, sans avoir à écrire un programme ad hoc.

5.3. La constitution des tableaux définitifs

La mise en forme et le stockage des tableaux définitifs résultent de l'exécution de deux opérations. La première a trait aux nomenclatures qu'il faut rendre compatibles d'un recensement à l'autre. Une étape DATA assure cette homogénéisation de chaque tableau : les diverses variables relatives à une nomenclature très désagrégée ou à des modalités de tableaux croisés sont regroupées en une seule nomenclature commune à tous les recensements.

La seconde opération, plus délicate, revient à agréger les nouveaux municipios, ceux qui sont apparus entre 1970 et 1980, de manière à exprimer les valeurs dans une maille administrative unique. PROC MEANS réalise cette opération selon le principe suivant :

```
PROC MEANS DATA=ENTREE SUM NOPRINT ; VAR V01-V99 ;
      OUTPUT OUT=SORTIE SUM=S01-S99 ;
      BY CODAGREG ;
```

Dans le tableau ENTREE figurent les observations d'origine, correspondant à une maille municipale différente pour chaque recensement ; le tableau SORTIE contient les observations agrégées selon une maille municipale unique où chaque observation est repérée par le code CODAGREG, variable supplémentaire du tableau des codes présenté plus haut.

5.4. Numérisation de la carte des municipios des cerrados

Une banque de données à base géographique doit être en mesure de restituer l'information sous forme de cartes. La carence quasi totale de l'EMBRAPA en informatique graphique n'a pas permis de réaliser des cartes thématiques. Cependant, le CPAC disposait d'un micro-ordinateur CANON AS-100 et d'une table à digitaliser japonaise (MUTOH) au grand format A0. Lorsqu'on a voulu numériser le fond de carte, cela n'a pas été possible immédiatement : ce matériel était inutilisable faute d'un progiciel adapté. Il a donc semblé utile de concevoir et de programmer en langage BASIC (le plus facilement utilisable sur CANON AS-100) un petit ensemble de programmes pour saisir le fond de carte.

Réalisé avec l'aide de Violette Cabos, ingénieur cartographe au CNRS, ce travail a donné lieu à l'élaboration d'un manuel pratique de numérisation publié en France (par le GIP RECLUS), dans sa version PC DOS. La figure 3 présente le résultat de cette numérisation qui prend la forme d'un fond de carte directement utilisable par SAS GRAPH sur les données de base.

6. CONCLUSION : COMPOSITION ET AVENIR DE SISECSO

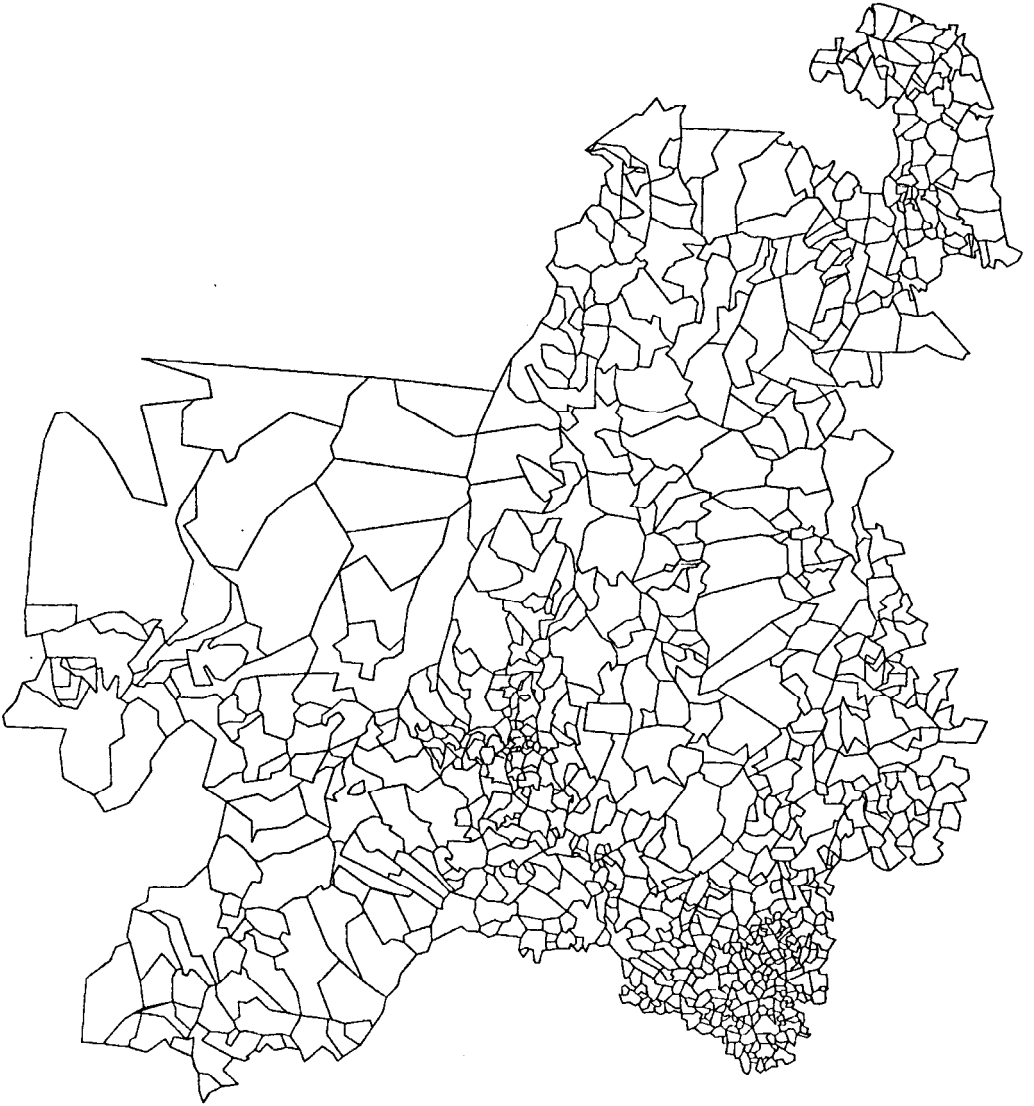
Au printemps 1987, SISECSO se compose de trois bases de données informatisées implantées sur le centre informatique de l'EMBRAPA.

La base de données agricoles comprend plus d'une centaine de variables pour chacun des recensements de 1970, 1975 et 1980. Ces variables sont strictement comparables et couvrent les domaines suivants : propriété de la terre, utilisation du sol, production agricole et élevage, taille des exploitations.

Les productions de l'agriculture et de l'élevage sont stockées dans une base de données spécifiques. Pour quelques 55 cultures, on dispose pour chaque année de 1977 à 1984, de la quantité produite, de la surface récoltée et de la valeur de la production ; pour l'élevage, on connaît la quantité et la valeur d'une quinzaine de productions.

La base de données démographiques, plus réduite, environ 80 variables, comprend les données des recensements de 1970 et 1980. Comme précédemment, ces variables sont directement comparables. Il s'agit à la fois de données strictement démographiques, comme le nombre d'habitants, les migrations ou la structure familiale, mais aussi d'informations sur le logement permettant de se faire une idée à la fois sur les habitants et sur leurs conditions de vie.

Figure 3



L'achèvement de la base de données 1970 - 1980 aura lieu fin 1987. Par la suite, les travaux pourront s'orienter selon trois directions : 1/ *l'exploitation scientifique* des données, 2/ *l'extension de la banque de données* à la totalité du Brésil, 3/ *l'alimentation en informations nouvelles*.

En l'absence de moyens informatiques graphiques suffisants, on n'est pas en mesure de faire un travail sérieux d'exploitation des données de SISECSO au Brésil. Aucune solution locale à court terme n'étant envisageable, il n'y a pas d'autre possibilité que de réaliser ce travail en France, sur l'un des systèmes du CNUSC ou du CIRCE. Il est regrettable que l'EMBRAPA ne puisse mettre à disposition les matériels et progiciels indispensables ; en leur absence, il est impossible de mener une recherche intéressante, au moment où elle devient réalisable, c'est-à-dire une fois acquises les sources statistiques.

Peut-être serait-il utile qu'un chercheur brésilien, intéressé par SISECSO, ait la possibilité de se former à l'analyse quantitative de l'espace en France ; une fois rentré au Brésil, il pourrait sans doute, plus efficacement que moi, convaincre ses interlocuteurs du bien-fondé de ces méthodes.

SISGEO est un projet couvrant la totalité du Brésil. En réalité, à part le module SISOLOS, tous les autres ne couvrent qu'une partie seulement du territoire brésilien. SISECSO n'échappe pas à la règle. En effet, chaque module est pris en charge par un centre de l'EMBRAPA, ce qui explique que SISECSO n'ait trait qu'aux seuls cerrados. Il est donc indispensable d'entreprendre un travail du même type, ailleurs au Brésil ; il n'y a que l'embarras du choix. Mais il serait vain de vouloir faire strictement la même chose et retomber ainsi dans un *"empirisme abstrait"* stérile. Il y a certainement d'autres questions qui se posent ailleurs, pouvant être abordées avec la même méthodologie quantitative.

A la fin de l'année 1987, le CPAC disposera d'une banque de données composée d'un peu plus de 3 000 variables, relevées sur 856 municipios. La forme sous laquelle ces données auront été stockées en ordinateur leur garantira la plus grande facilité d'accès. Mais ces données, relatives à la décennie 1970-1980, devront être complétées au fur et à mesure de la sortie de nouveaux recensements.

**ANNEXE : TRAVAUX DE PHILIPPE WANIEZ EN
RAPPORT AVEC SISECSO**

1/ *Les données et le territoire* (ouvrages techniques)

- Initiation au traitement informatique des données spatialisées. Paris, ORSTOM, coll. IDT, 119 p.

- Initiation à la numérisation pour la cartographie thématique. Montpellier, GIP RECLUS, 54 p.

- Initiation à l'analyse en surfaces de tendances. A paraître, 36 p.

2/ *Communications*

- Du système du technocrate à celui du géographe. VIIIème séminaire d'économie et de sociologie rurales. Montpellier, CIRAD/MESRU, 14-18 Septembre 1987

- Comment monter une base de données géographiques avec SAS. Contribution à la réunion annuelle des utilisateurs du système SAS. Chantilly, 21-22 Octobre 1987

3/ *Articles*

- Approche d'un grand espace : la maille municipale des cerrados brésiliens. Montpellier, GIP RECLUS, Mappemonde, n°3-1987, pp.38-41.

- Base de données pour la recherche géographique : la dynamique spatiale agricoles des cerrados au Brésil. Paris, ORSTOM, Cahiers de Sciences Humaines, à paraître.

Session 5 :
MODELISATION

**EXPERIENCE D'UTILISATION D'UN SUPER-
CALCULATEUR
POUR LA MISE EN PLACE D'UN MODELE
OCEANIQUE OPERATIONNEL**

Alain Morlière
(Océanographie Physique UR 101)

Université Pierre et Marie Curie
LODYC / Tour 14
4, place Jussieu
75252 Paris Cédex 5 France

RESUME - Les océans tropicaux jouant un rôle majeur dans les échanges énergétiques qui conditionnent l'évolution du climat de notre planète, il est apparu indispensable de pouvoir suivre en temps quasi-réel leur évolution de façon à améliorer la prévision climatique. Pour cela deux techniques complémentaires sont utilisées : l'une, ancienne, est basée sur l'observation systématique par navires marchands, elle se modernise avec l'usage des satellites ; l'autre, plus récente, repose sur la simulation numérique à l'aide de modèles.

L'objectif de notre travail est de fournir à partir des sorties d'un modèle numérique une description détaillée mensuelle de l'Océan Atlantique dans ses trois dimensions entre 20°S et 20°N. Le modèle utilisé a été mis au point par une équipe du LODYC ; il simule l'évolution de l'océan décrit comme un fluide en mouvement sous l'effet principalement du rayonnement solaire, des échanges de chaleurs liés aux changements d'état de l'eau et des échanges mécaniques dus aux vents. Les observations réalisées "in situ" seront assimilées dans le modèle pour en corriger les sorties.

Pour atteindre l'objectif visé, il faut maîtriser le fonctionnement du modèle dans un environnement de super-calculateur CRAY2, il faut maîtriser le flot des données nécessaires à l'excitation du modèle et enfin il faut définir et réaliser les produits terminaux les plus pertinents et les plus synthétiques. C'est cette expérience d'utilisation d'un super-calculateur que nous nous proposons d'exposer.

Dans ce genre d'opération, l'utilisation d'un ordinateur du type Cray1 (puis Cray2 à partir d'avril 1987) est indispensable, car seul ce type de calculateur permet de traiter d'énormes quantités d'information à travers un volume de calcul considérable.

Dans notre cas, la "grille" du modèle comporte 125 000 points auxquels sont attachés quatre paramètres qu'il faut conserver sur trois pas de temps, le tout en flottant double précision soit : 12 millions d'octets en mémoire (ou une gestion d'entrées-sorties très pénalisante). Les calculs qui permettent de passer d'un pas de temps à l'autre s'étalent sur plus de 4 000 lignes de Fortran. Pour traiter un mois de simulation il faut intégrer le modèle sur plus de 1 000 pas de temps. Le Cray2 avec ses 2 000 gigaoctets de mémoire centrale, sa capacité de vectorisation, ses quatre processeurs parallèles permet, moyennant quelques efforts de programmation, d'obtenir des temps de calcul tout à fait raisonnables.

En dehors de la partie proprement calcul, le reste est plus classique puisqu'il s'agit d'organiser, de valider des flots de données en entrée (notamment les données de vent issues du modèle atmosphérique du Centre Européen de Reading) ; en sortie, les produits graphiques sont élaborés à travers le logiciel NCAR.

INTRODUCTION

Depuis une dizaine d'années l'océanographie évolue dans le sillage de la météorologie et s'est de ce fait considérablement transformée avec le développement de l'intérêt porté à la climatologie et à la prévision du temps. Ceci est du en grande partie aux résultats qui ont montré le rôle très important joué par l'océan dans la redistribution de l'énergie thermique à la surface de la planète.

La distribution des flux méridionaux d'énergie transportée par la circulation de l'atmosphère et des océans à travers les parallèles (Figure 1) montre que le rôle de l'océan est globalement équivalent à celui de l'atmosphère et qu'il domine très nettement aux basses latitudes. C'est lui qui assure l'évacuation de la chaleur en excès aux faibles latitudes vers les hautes latitudes. L'océan, avec sa capacité thermique très importante et ses vitesses de déplacement lentes joue le rôle de "mémoire" à moyen terme des événements climatiques. C'est cette caractéristique qui laisse augurer d'un allongement de la prévision du temps par une meilleure connaissance de l'océan et des mécanismes qui le régissent.

Toutefois, dans le dispositif de prévision du climat, l'océan reste le point faible. Ceci est en grande partie du à la difficulté d'observations systématiques sur toute l'étendue maritime. Les satellites doivent fournir dans la prochaine décennie un élément de réponse à cette nécessité d'observations synoptiques systéma-

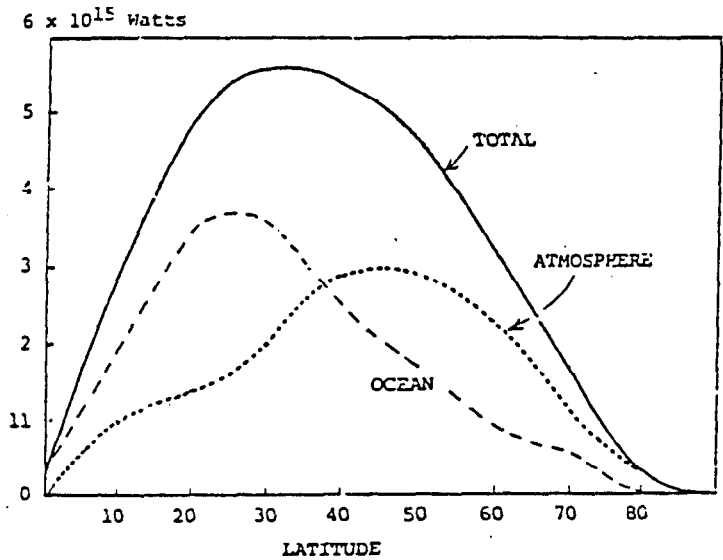


Fig. 1. Flux méridionaux d'énergie transportés par la circulation de l'atmosphère et celle des océans, à travers différents parallèles de latitude croissante.
(D'après Carrissimo, Oort et Von der Haar, 1985, J.P.O. 15).

tiques. Parallèlement à cette voie, il existe deux autres voies complémentaires : l'observation systématique par l'intermédiaire des bateaux marchands (qui, contrairement aux satellites permettent l'observation en profondeur) et la simulation de l'océan à partir de modèles numériques.

Les océanographes physiciens de l'ORSTOM, ont, jusqu'à présent, surtout travaillé dans le domaine de l'observation. Ils ont mis en place un réseau de mesures par navires marchands qui est actuellement la pièce maîtresse du dispositif d'observation permanent de l'océan tropical du programme international TOGA. Ce programme prévoit un accroissement de l'effort d'observation de cette région pendant dix ans ainsi qu'un développement de la modélisation océanique en vue d'une amélioration de la prévision climatique à des échelles de temps comprises entre le mois et l'année.

Quelques uns tentent depuis peu de prospecter la voie de la modélisation en utilisant des modèles numériques soit pour des études de processus, soit pour une simulation périodique la plus réaliste possible. C'est la voie que notre équipe a choisi. L'objectif de notre travail est de fournir, à partir des sorties d'un modèle une description détaillée mensuelle de l'océan Atlantique dans ses trois dimensions entre 20 N et 20 S. Pour obtenir la description la plus réaliste possible, nous prévoyons de contrôler les sorties mensuelles du modèle à partir des observations réalisées pendant le mois écoulé ; cette assimilation de données doit éviter les "dérives" dues : soit à des problèmes numériques, soit aux approximations faites lors de l'écriture du modèle. Il faut noter que l'assimilation des données dans les modèles océaniques est un problème mal connu qui fait l'objet de nombreuses recherches.

Une expérience analogue à la nôtre fonctionne aux Etats-Unis sur le Pacifique à l'aide d'un modèle différent ; de même, une équipe britannique doit prendre en charge l'océan Indien.

1. QU'EST-CE QU'UN MODELE OCEANOGRAPHIQUE ?

Pour construire un modèle numérique de l'océan, on part des équations de Navier-Stokes qui, pour un fluide en mouvement, expriment la conservation de la quantité de mouvement, de la masse et de la chaleur dans un repère lié à la terre. C'est un ensemble d'équations aux dérivées partielles non linéaires sur un domaine de géométrie complexe qu'il faut résoudre en tenant compte de conditions aux limites et d'un état initial donné. On fait un certain nombre d'hypothèses simplificatrices (hypothèse de Boussinesq, incompressibilité, hydrostatisme, mélange vertical convectif, "toit rigide",...). On précise les conditions aux limites (fond plat, frontières fermées, non glissement à la côte,..). De toute façon,

on ne sait pas résoudre de façon analytique le système obtenu d'où le recours aux méthodes numériques pour trouver des solutions approchées. On utilise la méthode des différences finies qui conduit à transformer le milieu physique continu en un milieu discontinu représenté par une grille découpant l'espace en portions élémentaires. Ceci permet de remplacer le système aux dérivées partielles par un système d'équations algébriques. Avec ces équations, en partant d'un état initial connu, on calcule les états successifs des différents paramètres du milieu : on *"intègre"* le modèle.

La discrétisation du milieu est choisie en fonction de ce que l'on veut étudier puisque les phénomènes ne peuvent être résolus que si le temps de propagation entre deux points de grille est inférieur au temps d'intégration d'où un équilibre entre la dimension des mailles, le temps d'intégration et la gamme de phénomènes étudiés.

Actuellement, il n'existe dans le monde que deux modèles tridimensionnels aux équations primitives de l'océan : un mis au point aux Etats-Unis, l'autre en France par une équipe du Laboratoire d'Océanographie DYnamique et Climatique (LODYC). Le modèle que nous utilisons est celui du LODYC mis au point par P.Decluse, P.Andrich, M.Chartier et M.A.Foujols.

La discrétisation a été faite suivant une grille en trois dimensions (110 x 72 x 16). En chaque point de la grille, on calcule quatre paramètres principaux et il est nécessaire de disposer de trois pas de temps pour procéder à l'intégration d'un nouveau pas de temps. D'où la nécessité de disposer à chaque pas de temps de 1.5 million de variables codées sur 8 octets soit environ 12 millions d'octets rien que pour les variables principales. Ces données doivent être stockées en mémoire si l'ordinateur utilisé en est capable, ou alors il faut passer par un stockage externe et une gestion complexe et pénalisante d'entrées sorties.

L'exécution représente un grand nombre de données à traiter à travers un volume très important de calculs ; seuls les nouveaux super-calculateurs sont capables d'absorber de tels travaux avec des délais d'exécution raisonnables. En effet les calculs qui permettent de passer d'un pas de temps à l'autre s'étalent sur plusieurs milliers de lignes de FORTRAN. Pour traiter un mois de simulation du modèle, il faut intégrer le modèle sur plus de 1000 pas de temps.

2. CONDITIONS DE REALISATION DE L'EXPERIENCE

2.1. Personnel

L'expérience a débuté en février 1987. Le personnel ayant travaillé sur le projet en 1987 comprend : J.Merle (10%), S.Arnault (10%) et A.Morlière (80%). L'apport d'un informaticien n'a pu être obtenu mais a été en partie compensé par un ingénieur vacataire pendant 3 mois en fin d'année et par un effort de formation des participants.

2.2. Matériels et logiciels

L'autorisation de travail sur le Cray passe par la soumission tous les six mois de projets auprès du Conseil Scientifique du Centre de Calcul Vectoriel pour la Recherche (CCVR) qui doit donner son accord et fixer le quota d'heures allouées au projet.

Le calculateur Cray pour la recherche est situé à Palaiseau, dans les locaux de l'Ecole Polytechnique. La puissance de ce calculateur réside à la fois dans sa formidable mémoire centrale (2000 Méga ou 2 Giga octets), dans sa capacité de vectorisation qui permet des gains de temps très importants sur toutes les opérations entre vecteurs (dans notre cas, le traitement vectorisé d'un pas de temps nécessite 3.04 secondes, non-vectorisé il en demande 80.94), dans ses quatre processeurs parallèles qui permettent soit le traitement de quatre tâches en même temps, soit le découpage des programmes en quatre morceaux exécutés parallèlement ce qui peut diviser les temps de séjour en machine par un facteur trois (le code que nous utilisons n'est pas encore parallélisé). La vectorisation et la parallélisation surtout, posent quelques problèmes de programmation qui ne sont pas simplifiés par le fait que le numéro de série du Cray français est le 6, c'est dire les quelques "bugs" à débusquer.

Pour utiliser le Cray 1 puis le Cray 2 (depuis avril 87), il faut transiter par une station frontale où sont stockés, édité les programmes et les fichiers de données et où sont récupérés les sorties graphiques. La station frontale utilisée est le Centre Inter Régional de Calcul Electronique (CIRCE) situé à Orsay et doté de deux calculateurs (IBM 3090/200 et NAS 9080). Le CIRCE est attaqué depuis le LODYC à Paris, à travers une grappe de terminaux IBM, un concentrateur et une ligne spécialisée. La station CIRCE est également utilisée à travers le réseau TRANSPAC à partir de micro-ordinateur personnel. Dans l'avenir, le réseau Ethernet-Sun du LODYC sera relié directement à celui du CCVR et de là, au Cray 2, d'où l'économie du passage au CIRCE.

Le système d'exploitation du Cray est UNICOS, succédané d'UNIX en traitement par lots. Sur le CIRCE, nous utilisons le sous-système TSO du système VMS/XA. Le langage de program-

mation est FORTRAN 77 (hors dessin le modèle comprend 4600 lignes de FORTRAN). Les bibliothèques utilisées sont celle de vectorisation d'UNICOS et celle de tracé : NCAR. Les sorties du modèle sont visualisées sur écran, papier (Versatec) et film de 35mm (pour archivage et reproduction en temps décalé).

Quelques chiffres relatifs à l'exécution des travaux sur le Cray : une intégration du modèle sur un mois demande environ 20 minutes de CPU et 21 Mégabytes de mémoire centrale. Pour mener à bien nos expériences avant de passer à la phase opérationnelle nous prévoyons plus de 100 heures CPU.

2.3. Données

Les données nécessaires au fonctionnement du modèle sont d'une part, les contraintes exercées par l'atmosphère à l'interface avec l'océan (vent et flux thermiques), d'autre part, les observations océaniques dont l'assimilation dans le modèle permet de rattraper les écarts éventuels à la réalité.

Les données vent ont, dans la phase préliminaire, deux origines. Elles proviennent du modèle de prévision climatique à moyen terme de Reading (Angleterre), elles sont acheminées par une ligne spécialisée vers la Météorologie Nationale à Paris qui les dispose en fichier sur le Cray à notre usage. Elles proviennent également des observations réalisées à bord des navires marchands et diffusées en temps réel sur le *Système Mondial de Transmissions* (SMT), elles doivent être sélectionnés, organisés après leur capture sur le SMT. Ces opérations sont encore dans leur phase expérimentale. Une fois disponible, le vent doit être contrôlé et mis à la maille du modèle avant d'être utilisé.

Les éléments des flux thermiques sont calculés à partir de séries climatologiques et ne posent pas de problème informatique. Ils posent par contre des problèmes physiques majeurs car de précision insuffisante.

Les données océaniques sont, pour l'instant, uniquement des profils thermiques verticaux entre la surface et 700m. Elles sont obtenues par les bateaux marchands le long des lignes de navigation et transmises par radio vers le SMT ou par balise Argos vers le Centre Argos de Toulouse. Nous les récupérerons par TRANSPAC auprès du Centre de données de sub-surface TOGA de Brest qui a la charge de regrouper et de valider ce type de données et devrait pouvoir nous les délivrer chaque mois dans des délais très courts.

2.4. Produits

Ce sont essentiellement des dessins sur film élaborés directement en sortie du Cray à partir du logiciel NCAR. Les produits restent à préciser mais ce sont principalement des planches à immersions fixes (Figure 2), des coupes est-ouest (Figure 3) et nord-sud (Figure 4) en fonction de la profondeur pour les principaux paramètres à savoir : température, vecteur courant et composantes du courant.

3. PREMIERS RESULTATS ET AVENIR

Les premiers résultats concernent le savoir faire acquis en matière d'utilisation de modèles océaniques et d'utilisation de systèmes informatiques modernes et performants.

Nous avons choisi de mettre au point nos méthodes de travail sur une année de référence (1984) particulièrement bien documentée en données océanographiques du fait de nombreuses opérations à la mer liées à l'expérience franco-américaine FOCAL SEQUAL. Nous avons procédé à une intégration du modèle sur toute l'année dans des conditions identiques à celles utilisées lors d'une intégration de l'expérience FOCAL (P.Deleduse), excepté le vent que nous avons calculé à partir des données d'observations élaborées par J.Servain. Ceci nous a donné des résultats différents (Figure 5), mais beaucoup plus proches de la réalité. En particulier nous retrouvons les remontées d'eaux froides à l'équateur en été qui avaient disparues dans l'intégration FOCAL utilisant un produit vent élaboré à partir des sorties du modèle atmosphérique de Reading. Ce résultat montre le rôle déterminant de la contrainte vent sur le réalisme des sorties du modèle et le soin qu'il faudra porter au choix de cette information.

Nous avons maintenant entrepris nos expériences de pseudo assimilation lors de réinitialisations périodiques du modèle à partir d'un état antérieur du modèle corrigé objectivement à l'aide des observations réalisées pendant la période d'intégration. Nous pensons tirer les conclusions des ces expériences vers la mi 1988 et entreprendre les premiers essais en temps réel fin 1988 afin d'apprécier exactement les conditions de mise en place d'un produit opérationnel.

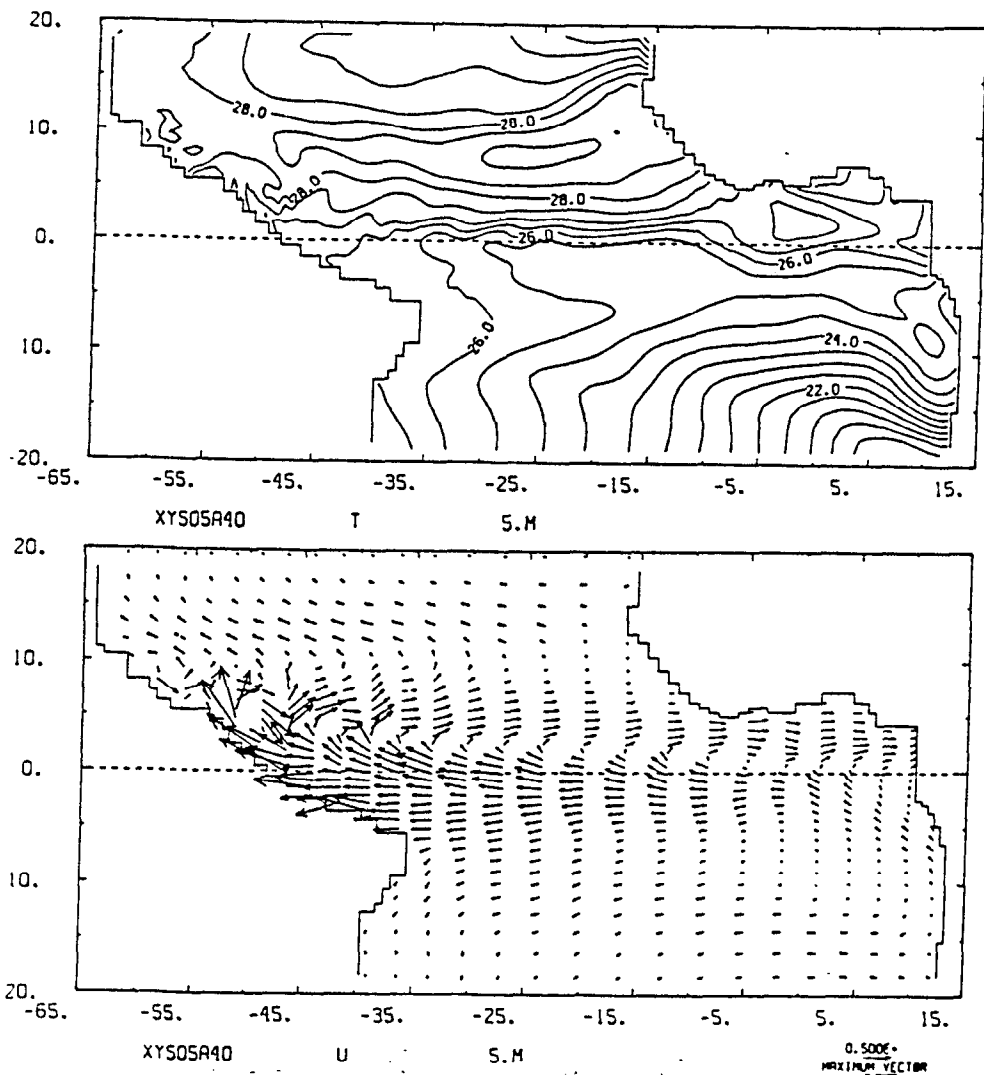


Fig. 2. Exemple sortie du modèle. Température et courant à 5M en octobre 1984.

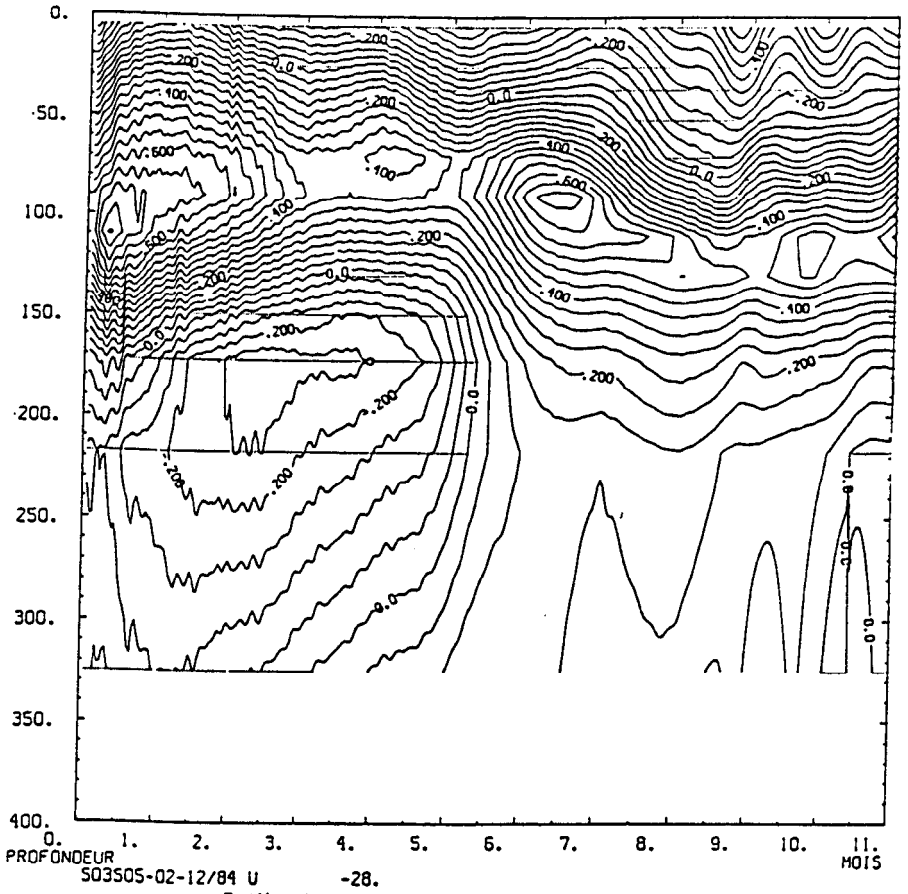


Fig. 3. Exemple de sortie du modèle. Evolution en fonction du temps de la composante zonale du courant sur l'équateur à 28° ouest.

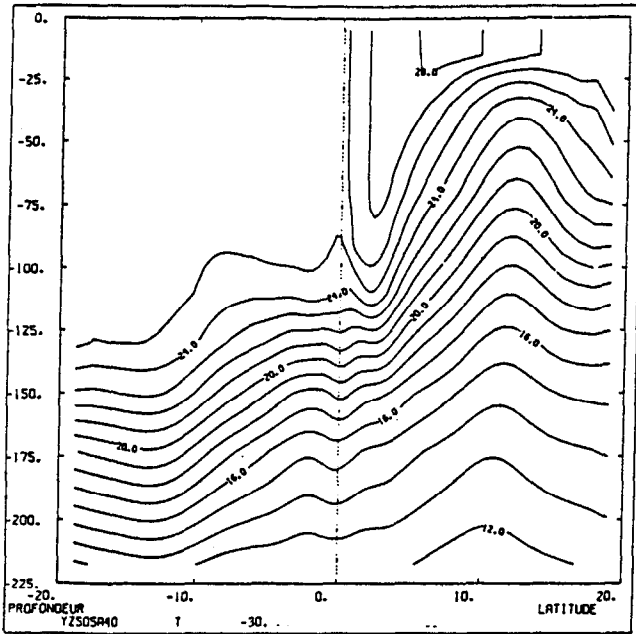


Fig. 4. Exemple de sortie du modèle. Coupe nord sud de température à la longitude 30° ouest.

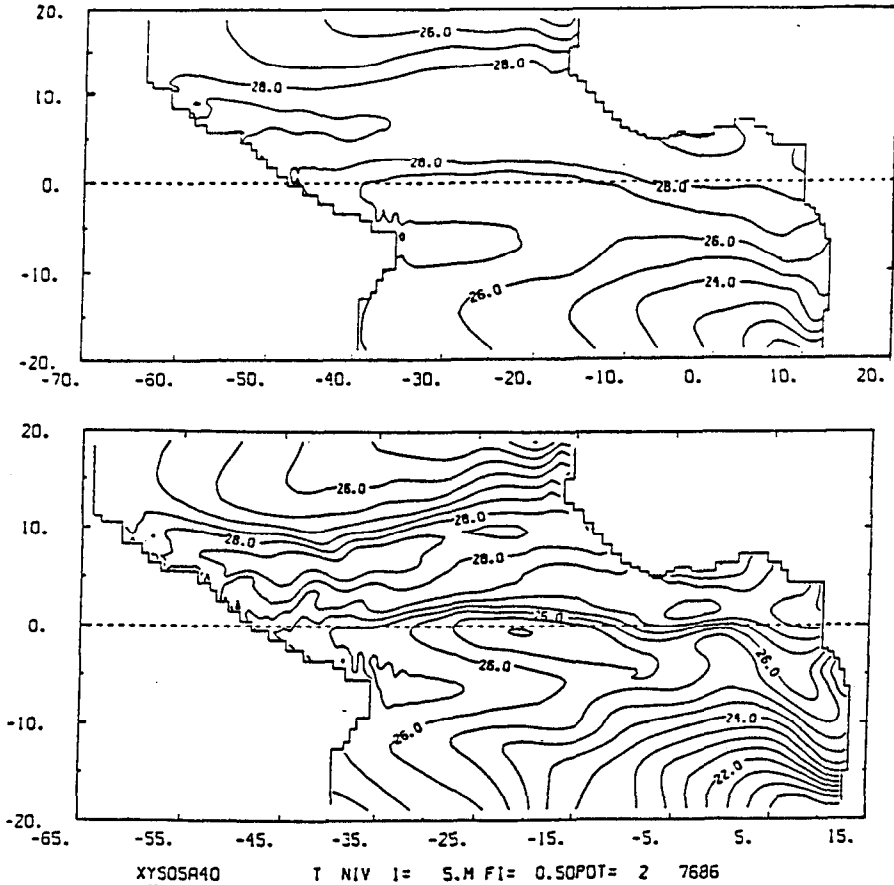


Fig. 5. Comparaison des sorties du modèle avec des vents différents au 1er août 1984.

a) Expérience FOCAL avec vent SEQUAL/FOCAL 2b.

b) Notre expérience avec vent Servain corrigé de la stabilité de l'air.

Remarque: - isolignes tous les 1° sur a)
 - isolignes tous les 0.5° sur b)

**Séminfor 1 : Premier Séminaire Informatique de l'ORSTOM
(Paris, 06-08/10/87)**

LISTE DES PARTICIPANTS

Nom & Prénom	Spécialité	U.R.	Lieu d'affectation
ALBOUY Yves	Géophysique	105	Bondy
ANTHONY François	Génétique	507	Montpellier
BENIZRI Claude	Informatique		
BEAUDOU Alain	Pédologie	203	Bondy
BOURRET Philippe		D.G	Paris
BRUNET JAILLY	Economie		Bamako
CANTRELLE Pierre	Epidémio.	709	Paris, ISD
CHAMPAUD Jacques	Géographie	406	Marseille
CHEVILLOTTE Hervé	Informatique	LIA	Bondy
COCHONNEAU Gérard	Info-Hydro.	502	Brasilia, Brésil
CORRE Daniel	Informatique		Brest
DANDOY Gérard	Géographie	502	Montpellier
DEJARDIN Jean	Biométrie	507	Bondy
DERUELLE	Cartographie		Bondy
FERRY Benoît	Démographie	709	Paris, ISD
FAURE Paul	Pédologie	104	Bondy
GIRARD Georges	Hydrologie	604	Fontainebleau
GUBRY Patrick	Démographie	801	Paris
GUISCAFRE	Hydrologie	L.H	Montpellier
HAMELIN Philippe	Démographie	205	Belém, Brésil
HENROTTE Marc			CNRS, Garchy
HOFF Michel	Botanique	201	Cayenne
LALOE Francis	Statistique		Paris, Orsay
LANGLOIS	Economie	203	Paris
LAPOUILLE André	Géophysique	106	Brest
LARDY Michel	Géophysique		Nouméa
LEBOZEC Jean-François	Informatique	SIG	Paris
LECHAUVE Jean-Jacques	Informatique	COB	Brest
LEGELEY Annick	Géophysique		Bondy
LORTIC Bernard	Téledétection	403	Bondy

LISTE DES PARTICIPANTS (suite)

MARTIN Jean-Yves			Paris
MICHAUX Monique	Informatique		Montpellier
MORETTI Christian	Botanique	707	Paris
MORLIERE Alain	Océanographie	101	Paris, LODYC
MORO Barthélémy	Informatique	U.F	Paris
MULLON Christian	Statistique	LIA	Bondy
PANSU Marc	Chimie	CMA	Bondy
PERROT Marie-Hélène	Documentation	HOR	Paris
PICHON Gaston	Biomath.	704	Bondy
PIRON Marie	Statistique	403	Ouagadougou
PONCET Yveline	Géographie	203	Bondy
RAMBAUD Dominique	Physi.Chimie	CS7	Bondy
ROUX Maurice	Statistique	CS7	Montpellier, CNRS
ROUX-FOUILLET Jean-Paul	Documentation	HOR	Bondy
SCHWARTZ André	Valorisation		Paris
SECHET Patrick	Informatique	502	Brasilia, Brésil
TABBAGH Jeanne			CNRS, Garchy
TREUIL Jean-Pierre	Informatique	SIG	Paris
VAUGELADE Jacques	Démographie	702	Ouagadougou
VERNEUIL Marie-Paule	Cartographie		Bondy
WANIEZ Philippe	Géographie	502	Brasilia, Brésil
ZENERINO Claude	Informatique	LIA	Bondy