

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

La gestion des registres du cancer sur microordinateur

Delfosse, Philippe; Goosens, Roselyne

Award date:
1983

Awarding institution:
Universite de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX (NAMUR)

INSTITUT D'INFORMATIQUE

LA GESTION DES REGISTRES DU
CANCER SUR MICROORDINATEUR.

Philippe DELFOSSE.

Roselyne GOOSSENS.

Promoteur : J.L. HAINAUT.

Mémoire présenté en vue
de l'obtention du titre
de Licencié et Maître en
Informatique.

Année Académique 1982 - 1983.

Avant-propos.

Il nous est impossible de citer ici les nombreuses personnes qui nous ont aidé dans ce travail.

Nous tenons à remercier les Docteurs Muir , Lévi, Ménégos, Parkin, Waterhouse; Messieurs Schiffers et Smans.

C'est à eux et à leur grande expérience en la matière que nous devons d'avoir été sensibilisés au rôle des registres du cancer dans la lutte contre ce fléau mondial.

Les conseils de Monsieur Hainaut, dans les choix techniques, nous ont été très précieux.

TABLE DES MATIERES.

	<u>pages</u> :
<u>Chapitre I</u> : Introduction.	1
<u>Chapitre II</u> : Exposé du problème.	6
2.0 Introduction.	7
2.1 Définition d'un registre du cancer.	8
2.2 Fonctions d'un registre.	11
2.2.1 L'input des informations.	11
2.2.2 Autres fonctions.	
2.2.2.1 Follow-up.	29
2.2.2.2 Edition de rapports.	29
2.2.2.3 Etudes statistiques.	30
2.2.2.4 Editions de listes.	36
2.2.2.5 Conclusion.	37
2.3 Les registres du cancer dans les pays en voie de développement.	38
2.3.1 Identification du patient.	38

2.3.2	Statistiques.	39
2.3.3	Sécurité de la base de données.	39
<u>Chapitre III</u> : Exposé de la solution.		
3.0	Introduction.	40
3.1	L'input des données.	43
3.2	Les autres fonctions.	45
3.3	Paramétrisation du système.	47
3.3.1	Solution au problème d'identification.	47
3.3.1	Les principe de paramétrisation est étendu à l'ensemble du système.	48
3.4	Les <u>fih</u> chiers.	49
3.4.1	La découpe ou fichier.	49
3.4.2	Contenu des fichiers.	49
3.4.2.0	Introduction.	49
3.4.2.1	Fichier-patient.	51
3.4.2.2	Fichier-tumeur.	53
3.5	Validation.	57

pages :

3.6 Choix du matériel. 58

3.7 Choix du langage. 61

Chapitre IV : Sous-système développés.

4.1 Gestion de la base de données. 64

4.1.0 Introduction. 64

4.1.1 Concepts de base. 64

4.1.1.1 Concept de la B.D. relationnel. 64

4.1.1.2 Terminologie utilisée. 68

4.1.2 Caractéristiques du S.G.B.D. 71

4.1.3 Description informelle. 72

4.1.3.0 Introduction. 72

4.1.3.1 Primitives de description de
la B.D. 73

4.1.3.2 Primitives d'exploitation de
la B.D. 75

4.1.3.3 Primitive de sécurité. 79

4.1.4 Description formelle. 82

	<u>pages</u> :
4.1.4.0 Introduction.	82
4.1.4.1 Primitives de description de la B.D.	82
4.1.4.2 Primitive d'exploitation de la B.D.	85
4.1.4.3 Primitives de sécurité.	124
4.1.5 Description technique.	127
4.1.5.0 Introduction.	127
4.1.5.1 Description de la structure des fichiers.	128
4.1.5.2 Description des variables et tableaux.	136
4.1.5.3 Transferts de données.	154
4.1.5.4 Description des algorithmes.	157
4.2 Gestion d'écran.	174
4.2.0 Introduction.	174
4.2.1 Concepts de base et nomenclature.	175
4.2.2 Fonctions de base.	176
4.2.3 Les primitives d'exploitation.	180

	<u>pages</u> :
4.2.4 Les primitives de description.	185
4.2.5 Implémentation.	191
4.2.5.1 Descripteur.	191
4.2.5.2 Primitives d'exploitation.	197
4.2.5.3 Primitives de description.	197
4.3 Lien avec la gestion de la base de données.	198
<u>Chapitre V</u> : Implémentation de la solution.	199
5.0 Introduction.	200
5.1 Structure des différents fichiers.	201
5.2 Paramétrisation.	201
5.3 Structure de l'application.	202
5.4 Implémentation des fonctions.	203
5.4.1 Pass.	203
5.4.2 Restart.	204
5.4.3 Follow.	204
5.4.4 Execfol.	205

5.4.5 Paramfol.	205
5.4.6 Editfol.	206
5.4.7 Lstfol.	206
5.4.8 Statis.	207
5.4.9 Rechpat.	207
5.4.10 Select.	209
5.4.11 Validata.	210
5.4.12 Lstselect.	210
5.4.13 Affselect.	211
5.4.14 Descbd et ecrbis2.	211
<u>Chapitre VI</u> : Conclusions.	212
6.0 Introduction.	212
6.1 Conclusions du test à Lausanne.	213
6.2 Nos critiques.	216
6.3 Choix du matériel.	218
6.4 Prolongements.	219
Bibliographie.	

CHAPITRE 1 : INTRODUCTION.

Introduction.

S'ingénier un jour de l'ouverture de la technique informatique au service de l'humanité ?

Domaine du possible ou rêve inaccessible ?

Ni l'un, ni l'autre, mais une réalité; car nulle personne avvertie ne songerait à contester encore le caractère utile, voire indispensable, de notre science auprès d'une autre, laquelle est toute entière vouée à la sauvegarde de la vie : la médecine.

L'on a pu discourir sur l'électronique à l'usage de la destruction, de l'armée et de la guerre; l'on a parlé à suffisance d'un ordinateur destructeur d'emploi, intégralement au service de la déshumanisation; tels sont les grands clichés qui s'imposent à la vue de l'homme non initié au fonctionnement de la technique ainsi qu'à ses utilisations, et qui constituent l'impression dominante qui s'écoule des propos que l'on peut échanger sur le sujet avec des profanes.

Il est vrai que tel perçu se lit à l'acuité d'une crise économique mondiale et à la lumière de nombreux conflits sporadiques durables ou non.

Peut être pourra-t-on un jour effacer du langage populaire une vision par trop dévastatrice de l'utilisation de notre modernisme et permettre de faire éclore la clarté que l'informatique peut dispenser sur la santé de l'homme et les progrès de la médecine.

Nous avons collaboré à la réalisation d'un pareil concours entre deux sciences, médecine et informatique; a priori peu enclines à se soutenir.

Notre travail de stage consistait en l'élaboration d'un logiciel de gestion des registres du cancer sur micro-ordinateur.

Lorsque l'on examine le dessein de ces registres qui est de mettre à jour les facteurs "favorables" à la maladie, point n'est besoin de démontrer ce qui se présente à l'évidence : les bienfaits du traitement automatique des données.

Mais l'horizon ne se ferme pas après notre réalisation, l'on peut évoquer subrepticement d'autres possibilités qui ont été exploitées : le traitement informatique des indices issus de l'examen scanner qui conduit à des propositions de diagnostic constitue un second exemple également suffisamment éloquent.

Alors, que conclure ?

Souhaiter une information dont le regard ne serait pas braqué sur une sphère tronquée en ce que axée sur la face négative du travail accompli par les générations successives d'ordinateurs.

Bien sûr il n'est pas facile d'écarter une idée reçue de l'esprit d'un homme traqué par le chômage et qui pour toute conclusion envisagera comme responsable de sa situation "une machine source de déroute", pas plus

qu'il n'est aisé de faire la démonstration dans un esprit peu averti en ce domaine que pareil traitement réalisé dans le cadre d'un registre du cancer n'est pas au demeurant un service de statistiques; l'atmosphère de récession favorisant la culture de tels ressentiment. Mais nous pouvons quand même essayer.

Nous avons développé ce travail dans le cadre d'un stage effectué au Centre International de Recherche contre le Cancer à Lyon.

Cet organisme dépend de l'Organisation Mondiale de la santé.

Le Centre possède deux unités principales vouées au service d'une même cause, la lutte contre le cancer, d'une part l'unité de recherche en laboratoire et d'autre part un service d'épidémiologie et biostatistiques. C'est au sein de ce dernier que nous avons accompli notre tâche.

Les périples de nos travaux nous ont conduits à tester notre réalisation au registre Vaudois des tumeurs de Lausanne.

Les perspectives sont de tester, dans les prochains mois, le logiciel ainsi élaboré dans un pays d'Afrique du Nord.

CHAPITRE 2 : EXPOSE DU PROBLEME.

2.0 Introduction.

Les pays en voie de développement disposent déjà de registres du cancer, lesquels sont pour la plupart attachés à un hôpital mais sans par ce fait relever nécessairement de la catégorie des registres dits "hospitaliers" (cfr. infra 2.1).

Le traitement des données y afférentes est manuel.

Le souhait du C.I.R.C. est de multiplier au maximum l'ouverture de registres dans le monde entier. Ceci peut paraître paradoxal à la lumière de la connaissance du problème majeur des pays du tiers monde qui n'est pas celui du cancer (°); mais les données concernant les cas de tumeurs cancéreuses dans ces contrées sont d'une grande utilité pour servir les buts poursuivis par l'organisation. En effet, on peut étudier l'impact du cancer dans un groupe ethnique donné, de par son mode de vie et mettre en évidence des facteurs favorisant l'apparition de la maladie.

Notre travail consiste dès lors en l'élaboration d'un système d'aide à la gestion des registres et ce sur microordinateur, l'ensemble de l'opération reposant sur un budget de 10.000 \$ jugé acceptable pour ce type de registre. (°°)

(°) en effet, la population meurt avant d'avoir atteint l'âge moyen auquel on contracte la maladie ce qui relègue la question aux accessits.

(°°) Le système n'est pas destiné à de gros registres : il s'agit d'enregistrer de l'ordre de 2.000 cas par an.

Nous précisons que le système est destiné à des personnes qui n'ont reçu aucune initiation à l'informatique.

En proposant ce système, le C.I.R.C. espère imposer une certaine standardisation, en vue de faciliter les études comparatives ultérieures .

Nous montrerons au cours de cet exposé combien cette standardisation est incomplète malgré le recours à la codification selon des critères internationaux.

Afin de déterminer les fonctions que notre système doit prendre en charge, dans une première phase de l'exposé nous présenterons les registres du cancer en les définissant et en les répertoriant; puis, nous en montrerons le fonctionnement général et nous passerons en revue les autres fonctions réalisées; enfin, nous soulignerons les problèmes spécifiques aux registres dans les pays en voie de développement.

2.1 Définition d'un registre du cancer.

Le cancer reste, dans les pays occidentaux, la deuxième (ou troisième) cause de décès (suivant le pays), après les maladies cardiovasculaires.

Les registres du cancer jouent un rôle primordial dans la lutte contre ce fléau : les recherches en laboratoires n'ont pas apporté, à ce jour, une solution globale au problème de cette maladie; on essaye donc de déterminer non les causes, sinon

le problème serait résolu, mais les facteurs susceptibles de favoriser l'apparition de la maladie (exemple : tabac et cancer des voies respiratoires).

Les registres du cancer tentent d'évaluer l'impact du cancer sur la population, ils collectent, enregistrent et analysent les données sur des cas individuels de maladie cancéreuse.

Il s'agit de rassembler des informations aussi exactes et complètes que possibles concernant tant le patient que sa tumeur. A la longue se crée une base de données importante, et l'on essaye d'en dégager des facteurs étiologiques (°).

Voici un exemple concret (registre de Birmingham - Angl.):

on a constaté chez les ramoneurs de cheminées ainsi que chez les tourneurs un même type de cancer.

On a cherché alors les facteurs communs à tous ces individus : ces personnes étaient continuellement en contact avec des huiles et autres dérivés; et de là on a supputé qu'il s'agissait de la cause de la maladie.

Dès lors, on a obligé ces personnes à porter des vêtements protecteurs pendant leur travail; et le mal a pratiquement disparu.

(°) L'étiologie cherche à établir les causes des maladies; ces causes peuvent être déterminantes (directes) par exemple les virus, bactéries, .. ou favorisantes (conditions de vie, hérédité,...).

Il existe deux types de registre du cancer : les registres hospitaliers et les registres 'basés sur une population'; nous appellerons ces derniers registres de population.

Les registres de population étudient la population d'une région géographiquement bien définie, ce qui est rarement le cas pour les registres hospitaliers qui comme leur nom l'indique sont attachés à un hôpital. Un hôpital est installé dans une région, mais la population hospitalière n'est pas nécessairement limitée à celle-ci, l'étalement de la fréquentation étant lié à la réputation de l'hôpital et au perfectionnement des équipements.

Les sources de données sont différentes; les registres de population reçoivent leurs informations des hôpitaux, laboratoires et des médecins, ils utilisent aussi les certificats de décès, les dossiers de sécurité sociale, les fichiers de mortalité ...; alors que les sources d'un registre hospitalier émanent des différents services de l'hôpital (chirurgie, radiothérapie,...).

Les deux types de registre se différencient également par les buts poursuivis par chacun d'eux : la finalité des registres hospitaliers est d'évaluer l'efficacité des traitements au travers d'études sur la population hospitalière; alors que, les registres de population procèdent au calcul de l'incidence par rapport à la population de base (^o), des taux de sur-

(^o) On entend par là, la population couverte par le registre.

vie..., et
essayent de déterminer les facteurs et populations à
risque ainsi que d'évaluer les effets de la prévention
et des programmes de détection.

Une dernière différence remarquable entre ces 2 catégories de registres se situe au niveau du follow-up (°); le follow-up des registres hospitaliers est actif; celui des registres de population est beaucoup plus difficile à réaliser; il est indirect et se fonde sur la bonne volonté des hôpitaux, des médecins ... à leur fournir les indications nécessaires.

2.1 Fonctions d'un registre.

2.1.1 L'input des informations.

Remarque préliminaire : que le traitement des données soit manuel ou automatisé, il existe un dossier par patient.

Par dossier, nous entendons une chemise dans laquelle on placera tous les documents reçus concernant ce malade et éventuellement des fiches reprenant les informations essentielles concernant la personne et sa(s) tumeur(s).

Les documents arrivant au registre sont examinés du point de vue de la cohérence des données et de leur complétude.

(°) follow-up=suivi du patient. Ce point sera approfondi ultérieurement.

On regarde si les informations concernent un patient connu ou non du registre. Cette phase est appelée le 'RECORD LINKAGE'.

S'il s'agit d'un nouveau cas, celui-ci est enregistré après codification des données, sinon l'enregistrement est mis à jour.

Détaillons les grandes étapes de la saisie des données.

Record linkage.

Les informations arrivent au registre sous des formes diverses (rapports d'anapathologie, certificat de décès, ...)

On peut recevoir, concernant un même patient, des données de sources différentes. Il faut recouper ces éléments entre eux et avec les fichiers existants. A cet effet, presque tous les registres tiennent à jour un fichier signalétique des patients. Ce fichier est manuel, il constitue un index. Ceci constitue une phase primordiale : il s'agit en effet d'éviter des enregistrements multiples concernant un même malade afin de minimiser les erreurs statistiques.

Traitement patient inconnu.

On ouvre un nouveau dossier pour ce patient. Certaines données telle la localisation de la tumeur sont codifiées suivant des normes internationales.

Malgré cela une standardisation complète n'est pas atteinte. Les médecins interprètent différemment les rapports médicaux; ainsi une tumeur de la peau à proximité immédiate des lèvres pourra être interprétée par certains comme un cancer de la peau, par d'autres comme cancer de la lèvre.

Néanmoins, pareille divergence n'a pas d'influence notable sur l'interprétation des résultats. Mais les registres travaillent avec des versions différentes des listes de codes. Ceci apporte par mal de difficultés, lors d'études comparatives des résultats, des conventions de codes sont nécessaires et ces dernières ne sont pas bien définies.

Les principales informations concernant la personne et sa tumeur sont le plus souvent reprises sur une fiche de notification (exemple : fig. I et II) qui est jointe au dossier. Certains registres (automatisés) négligent cette phase et enregistrent directement le cas, la fiche faisant redondance avec l'enregistrement sur support magnétique.

Les données sont validées; exemple : contrôle croisé sur la localisation de la tumeur et le sexe du patient.

DONNEES PERSONNELLES

cc/no _____

Situation à la date de référence, le :		cc=3
Nom:	_____	8-
Nom de jeune fille:	_____	
Prénoms: _____	<input type="checkbox"/> masc. <input type="checkbox"/> fém.	48-
Prénoms du mari: _____		
Né(e) le: _____ jour _____ mois _____ année		
A: _____	Canton/pays: _____	
Si né à l'étranger: année d'arrivée en CH (GE): _____		
Etat civil: <input type="checkbox"/> célib. <input type="checkbox"/> marié <input type="checkbox"/> veuf		
<input type="checkbox"/> séparé <input type="checkbox"/> divorcé <input type="checkbox"/> SR		
Nationalité (canton/pays): _____		
		cc=4
Adresse à Genève: _____	} No rue	8-
_____		} No post
Adresse permanente (si non domicilié): _____	} Ville	43-
_____		} Pays
Profession individuelle: _____		

Profession du chef de famille: _____		

Changements de nom et d'adresse (MAJ).	
_____	} dès le : _____

_____	} dès le : _____

_____	} dès le : _____

fig.1a

FORMULE D'ENREGISTREMENT

cc/No _____

DONNEES PERSONNELLES		cc=1
SEXE:	1 <input type="checkbox"/> masc. 2 <input type="checkbox"/> fém. 3 <input type="checkbox"/> SR	8
ETAT CIVIL:	1 <input type="checkbox"/> cél. 2 <input type="checkbox"/> mar. 3 <input type="checkbox"/> veuf 4 <input type="checkbox"/> sép. 5 <input type="checkbox"/> div. 6 <input type="checkbox"/> SR	9
NE LE	____ j. ____ m. ____ a.	10-
A (canton/pays):	_____	17-
SI A L'ETRANGER: ANNEE D'ARRIVEE EN CH (GE):	_____	20-
NATIONALITE (canton/pays):	_____	23-
DOMICILE LEGAL:	1 <input type="checkbox"/> GE 4 <input type="checkbox"/> CH 2 <input type="checkbox"/> NE 5 <input type="checkbox"/> France 3 <input type="checkbox"/> VD 6 <input type="checkbox"/> a. pays 7 <input type="checkbox"/> SR	26
PROP. CHEF DE FAMILLE:	1 <input type="checkbox"/> ouv. 2 <input type="checkbox"/> empl. 3 <input type="checkbox"/> cadre 4 <input type="checkbox"/> SR	27
PROP. INDIVIDUELLE:	_____	28-

DIAGNOSTIC		cc=1
LOCALISATION:	_____	61-
Organe, point de départ, côté, etc.:	_____	
EXTENS. TUM. PRIMAIRE:	1 <input type="checkbox"/> In Situ 2 <input type="checkbox"/> local. 3 <input type="checkbox"/> ext.rég. 4 <input type="checkbox"/> à dist 5 <input type="checkbox"/> SR	65
Fixation, etc.:	_____	
DIMENS. TUM. PRIMAIRE:	_____	66-
ADENOPATHIE REG.:	1 <input type="checkbox"/> + 2 <input type="checkbox"/> - 3 <input type="checkbox"/> SR	68
Nombre, local, etc.:	_____	
METASTASES A DISTANCE	1 <input type="checkbox"/> + 2 <input type="checkbox"/> -/SR	69
Localisation:	_____	
TYPE HISTOLOGIQUE	_____	70-
DIFFERENTIATION	_____	75

ANAMNESE		cc=1
ORIGINE DIAGNOSTIC:	1 <input type="checkbox"/> consult. sur sympt. 2 <input type="checkbox"/> check-up 3 <input type="checkbox"/> découverte fortuite lors de soins 4 <input type="checkbox"/> autopsie 5 <input type="checkbox"/> décès (sans autops.) 6 <input type="checkbox"/> SR	32
SYMPTOMES PREEXISTANTS:	1 <input type="checkbox"/> oui 2 <input type="checkbox"/> non 3 <input type="checkbox"/> SR	33
Lesquels:	_____	
RANG DE LA TUMEUR: (SR = 0)	_____	34
LOCALISATION DE LA TUM. PRECED.:	_____	35-

TRAITEMENTS (radicaux ou palliatifs agissant directement sur la tumeur ou les métastases)		cc=1
CHIRURGIE:	effectuée { 1 <input type="checkbox"/> radicale 2 <input type="checkbox"/> palliative 3 <input type="checkbox"/> sans précision 4 <input type="checkbox"/> prévue 5 <input type="checkbox"/> non envisagée 6 <input type="checkbox"/> SR	76
RADIOTE.:	1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> SR	77
CHIMIOTH.:	1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> SR	78
AUTRES:	1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> SR	79
Lesquels:	_____	
DEBUT TRAITEMENT	____ m. ____ a.	8-

BASE DU DIAGNOSTIC		cc=1
STATUS CLINIQUE:	1 <input type="checkbox"/> +/s 2 <input type="checkbox"/> - 3 <input type="checkbox"/> SR	39
RADIODIAGNOSTIC:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	40
CATSCAN:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	41
ENDOSCOPIE:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	42
ST. OPERATOIRE:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	43
CYTO/HEMATO:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	44
BIOPSIE:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	45
AUTOPSIE:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	46
CERT. DECES:	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	47
AUTRES	1 <input type="checkbox"/> " 2 <input type="checkbox"/> " 3 <input type="checkbox"/> "	48
Lesquelles:	_____	
DATE DE L'EXAMEN LE PLUS PROBANT:	____ j. ____ m. ____ a.	49-
DATE REF.:	____ j. ____ m. ____ a.	55-

MEDECINS ET SERVICES SOIGNANTS		cc=2
INSTANCE CHARGEE DU 1ER TRAIT	_____	12-
MEDECIN CHARGE DU P-U	_____	16-
SERV. HOSP. CHARGE DU P-U	_____	20-

fig. 1b

SOURCES D'INFORMATION			cc=2
PATHOLOGIE	1 <input type="checkbox"/> oui	2 <input type="checkbox"/> non	24
CYTOLOGIE	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	25
RADIOLOGIE	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	26
A. CENTRES OFF. DE DIAGNOSTIC	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	27
ONCC-HEMATO	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	28
RADIOTHER.	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	29
A. SERVICES SOIGNANTS HC	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	30
BEL AIR	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	31
GERIATRIE	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	32
A. SERVICES SOIGNANTS OFF.	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	33
CERTIFICAT DECES	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	34
A. REGISTRES	1 <input type="checkbox"/> "	2 <input type="checkbox"/> "	35

SOURCES D'INFORMATION (suite)		cc=2
MED. PRAT. A	_____	36-
MED. PRAT. B	_____	40-
MED. PRAT. C	_____	44-

VARIABLES SUPPLEMENTAIRES		cc=2
T	_____	48-
N	_____	50-
M	_____	52-
Code TNM	_____	53-
Conf. ADENO	1 <input type="checkbox"/> + 2 <input type="checkbox"/> - 3 <input type="checkbox"/> SR	54-

OBSERVATIONS:

Date:

fig.1c

Visa:

		1	2	3	4	5	6	7		
		Type	Registry Number			Year				
INDIAN CANCER SOCIETY AND TATA MEMORIAL HOSPITAL PAREL, BOMBAY-12.		BOMBAY CANCER REGISTRY								
Hospital _____		Case No. _____								
Referred by _____		Date _____								
Name _____		(First name)		(Middle name)			(Surname)			
Sex : 1. Male () 2. Female ()							S E X	8		
Religion : 1. Hindu () 2. Muslim () 3. Christian () 4. Parsi () 5. Buddhist () 6. Jain () 7. Sikh () 8. Jew () 9. Others ()							R E L I G I O N	9		
Caste : _____ Language : 1. Marathi () 2. Gujarathi () 3. Others () 9. Not known ()							L A N G U A G E	10		
Actual age : _____ Birth date : _____ (in years)							A G E	11 12		
Marital status : 1. Unmarried () 2. Married () 3. Widowed () 4. Divorced () 5. Separated () 9. Not known ()							M A R I T A L S T A T U S	13		
Income : 1. Less than Rs. 50 () 2. Rs. 50-99 () 3. Rs. 100-349 () (per month) 4. Rs. 350-749 () 5. Rs. 750-999 () 6. Rs. 1000 + 9. Not known ()							I N C O M E	14		
Mother tongue : _____ Education : _____										
Size of family : Males _____ Female _____ Earning members _____										
A D D R E S S	Bombay : C/o _____							W A R D C I R C U L A R	15	
	House/Chawli _____ No. _____								16	
	Near _____ Street/Road _____								17	
Room No. _____ Floor _____ Bombay district _____ No. _____							18			
Ward : _____							19			
Native place : At _____ Post : _____							S T A T E	20		
Tal : _____ District : _____								21		
State : _____										
R E S I D E N C E	Past : 1 _____ Duration _____ Years							R E S I D E N C E	22	
	2 _____ " _____ "								23	
	Present duration in Greater Bombay _____ "								24	
1. Bombay () 2. Non-resident () 3. Not known ()							25			
1. New cancer case () 2. Diagnosed earlier () 3. Suspected cancer () 4. Non-cancer ()										

fig.IIa

O C C U P A T I O N	Past : 1.....	Duration	Years	O C C U P A T I O N		22
	2	"	"			23
	3	"	"			24
	Present:	"	"			
H I S T O R Y	Previous cancerous disease	Yes () Site		S Y M P D U R A T I O N		25
		Treatment				26
		Date of diagnosis				27
		No ()				
	Present symptoms.....			C A F A M		28
	Duration of symptoms :					
	Treatment taken during this period :			C I R C U M C I S I O N		29
	Cancer in family : Relationship to the patient					30
	Site of the cancer					31
	Circumcision (In case of women - husband's) : If yes, Reason : 1. Religious () 2. Health () 3. Unknown () 8. No () 9. Not known ()					
	At what age:					

Additional information :

Attendance at the Tata Hospital { Yes () Hospital Registration No. _____
Treatment received : _____
No ()

Name of the Social worker _____

Date of interview _____

fig.IIb

Registry Number

D	Primary site	Primary site	
I	If primary unknown } Main secondary }		
A	Date of Diagnosis :	Date	Month
G	Date of recurrence :		
N	Extent of Involvement	1. In situ () 2. Confined to organ ()	Involvement
O		3. Metastases to regional glands () 4. Penetration to adjacent tissues ()	
S	Method of diagnosis	1. Biopsy () 2. Cytology () 3. Autopsy with histology ()	Method
I		4. Blood studies () 5. Operation findings () 6. Autopsy without histology ()	
S	Histology report :	Grade :	Histology
I	Reported by :	Path No. :	
T	Type :		Type
R	1. Surgery alone () 2. Irradiation alone () 3. Chemotherapy alone ()		
E	4. Surgery + Irradiation () 5. Surgery + Chemotherapy () 6. Irradiation + Chemotherapy ()		
A	7. Surgery + Irradiation + Chemotherapy () 8. Other () 9. No treatment ()		
T	Surgery: 1. _____ Date _____ Done at _____		Surgery
M	2. _____ " _____ " _____		
E	3. _____ " _____ " _____		
N	4. _____ " _____ " _____		
T	Irradiation _____ Date _____ Done at _____		Irradiation
	_____ " _____ " _____		
	_____ " _____ " _____		
	Other : _____ Date _____ Done at _____		
	_____ " _____ " _____		
No Treatment	1. Treated previously () 2. Treated elsewhere () 3. Extensive disease ()		No treatment
	4. Old age () 5. Weak patient () 6. Refused by the patient ()		
	7. Referred to other Hospital () 9. Not known ()		

Date of discharge :

Follow up notes :

fig.IIc

F I N A L D I A G N O S I S	Final diagnosis				
	Description of tumour				
	Involvement				
	Nodes				
	Metastases				
H I S T O R Y	Date	Biopsy Number	Specimen from	Report	Reported by
O P E R A T I O N	Post operative diagnosis:				Date:
	Findings:				
X R A Y	Location:				Date:
	Report				
	Ba meal/swallow Report:				Date:
B L O O D	Blood Group:				Date:
	Studies:				
C Y T O L O G Y	Specimen from:				Date:
	Report:				
A U T O P S Y	Report:				Date:
Date of death					52- 57
Cause:					58- 61
Reported by:					62- 64
1. Hospital () 2. Private practitioner () 3. Other ()					
9. Not known ()					

fig.IId

	3: 2	Registry number.....	1-7	
Teeth cleaning:	1. Dantan twig () 2. Paste () 3. Powder () 4. Dung ash ()	T E E T H	8	
	5. Charcoal powder () 6. Burnt tobacco () 7. Others ()			
	8. No () 9. Not known ()			
Instrument:	1. Brush () 2. Dantan twig () 3. Finger () 4. Other ()	I N S T R.	9	
	8. No () 9. Not known ()			
Tongue cleaning if yes	1. Metal cleaner () 2. Plastic cleaner () 3. Other cleaner ()	T O N G U E	10	
	4. Dantan twig () 5. Finger () 6. Other () 7. Occasionally ()			
	8. No () 9. Not known ()			
Use of Dentures:	If yes, does it hurt? 1. Yes () 2. No () 3. Unknown ()	H U R T S	11	
	8. No () 9. Not known ()			
If yes, how long in years		Duration	Missing	12 to 15
	If no, number of missing teeth			
C H E W I N G	If yes, 1. Spitting () 2. Swallowing () 3. Unknown () 4. Occasionally ()	Y E S	16	
	8. No () 9. Not known ()			
	1. Pan () 2. Betel nut () 3. Tobacco () 4. Pan & Betel nut ()			
E T H E R	5. Pan & Tobacco () 6. Tobacco & Betel nut () 7. Pan + Betel nut & Tobacco ()	T Y P E	17	
	8. Occasionally/No () 9. Not known ()			
F R E Q U E N C Y	Use of lime: Yes () No () Other ingredients:			18 to 21
	Frequency.....Duration.....in years	Frequency	Duration	
	Type of tobacco used			
A M O U N T	Amount spent per day: Rs.....Ps.....	Amount Spent	Duration quid	22 to 25
	Quid chewing, how long?.....in minutes		Quid in sleep	
	Quid placement during sleep: 1. Yes () 8. No () 9. Not known		Quid Placement	
A G E	Quid placement: 1. Left buccal fold () 2. Right buccal fold ()			26 to 27
	3. Labial fold () 4. Other ()..... 9. Not known ()		Age at start	
	Age started chewing:			
S M O K I N G	If stopped chewing, When?			28
	If yes, Inhalation: 1. Yes () 2. No () 3. Inhalation-Unknown ()	Y E S	29	
	4. Occasionally () 8. No () 9 Not known ()			
1. Bidi () 2. Cigarette () 3. Hukka () 4. Chillum () 5. Cigar ()	T Y P E			30
6. Pipe () 7. Chiroot () 8. Other				
B R A N D	Brand:.....			31 to 34
	Frequency: 1..... Duration.....Years	Frequency	Duration	
	2..... "			
A M O U N T	Amount spent per day: Rs.....Ps.....	Amount	Age at Start	35 to 38
	Age started smoking.....			
	If stopped smoking, when?			

fig.IIe

S N U F F	1. Yes () 2. Occasionally () 8. No () 9. Not known ()															
	Frequencytimes per day										Yes		Frequency		Duration	
Duration: Foryears																
A L C O H O L	If yes { 1. Occasionally () 2. Foreign brew ()										Yes		Quantity		Duration	
	3. Local brew () 4. Type-not known ()															
8. No () 9. Not known ()																
Quantity per dayfor..... years																
1. Casual () 2. Moderate () 3. Heavy () 9. Not known ()																
F O O D	1. Vegetarian () 2. Non-vegetarian () 9. Not known ()										Veg.		Temp.		Spicy	
	1. Hot () 2. Moderate () 3. Cold () 9. Not known ()															
1. Spicy () 2. Moderate () 3. Non spicy () 9. Not known ()																
History of trauma :																
For women only										Age at Marriage: First:Second:		A G E				
If widowed or divorced: How long?.....Years																
Age at first delivery.....Age at last delivery.....										First		Last				
Number of { Pregnancies:.....Abortions.....Still Births										Preg.		Ab.		S.B.		
Dead Children.....Living Children.....																
S p a c i n g o f P r e g n a n c i e s	Number										Average		Spacing			
	1															
	2															
3																
4																
5																
6																
7																
8																
9																
Age of Mother																
Age at death of the child																
Breast Feeding: If yes, 1. Left side () 2. Right side () 3. Both sides () 4. Side not known ()										Yes		How long				
How long?.....																
8. No () 9. Not known () Reason.....																
Age at Menstruation:										Menstruation		Menopause				
Age at Menopause:																
Leucorrhoea: Yes () No () How long?																
Menstruation: 1. Regular () 2. Irregular () 9. Not known ()										Reg.		Pain		Scanty		
1. Painful () 2. No pain () 9. Not known ()																
1. Scanty () 2. Profuse () 9. Not known ()																
Has menstruation been stopped artificially: { If yes, Reason,										Yes						
1. Family planning () 2. Menstrual trouble () 3. Reason-unknown ()																
8. No () 9. Not known ()																
Method. 1. Radium () 2. Surgery () 3. X-Rays () 9. Not known ()										Method.						
At what age:																
Occupation of Husband/Guardian:																

fig.IIf

Des compléments d'informations peuvent être demandés soit pour correction des données après validation soit qu'elles soient jugées incomplètes (bouclage dans la fig. III). Le dossier est classé dans un fichier en attente.

Traitement patient connu.

On met éventuellement à jour les informations propres au malade (exemple : changement d'adresse, ...).

Deux possibilités :

les données concernent une tumeur déjà connue du registre. Dans ce cas une simple mise à jour suit; dans le cas contraire on ajoute une nouvelle fiche au dossier.

La figure III reprend les grandes étapes de la saisie des données.

Les détails de la procédure seront exprimés au travers des figures IV, a, b, c et d, qui parlent d'elles mêmes et ne suscitent aucun commentaire.

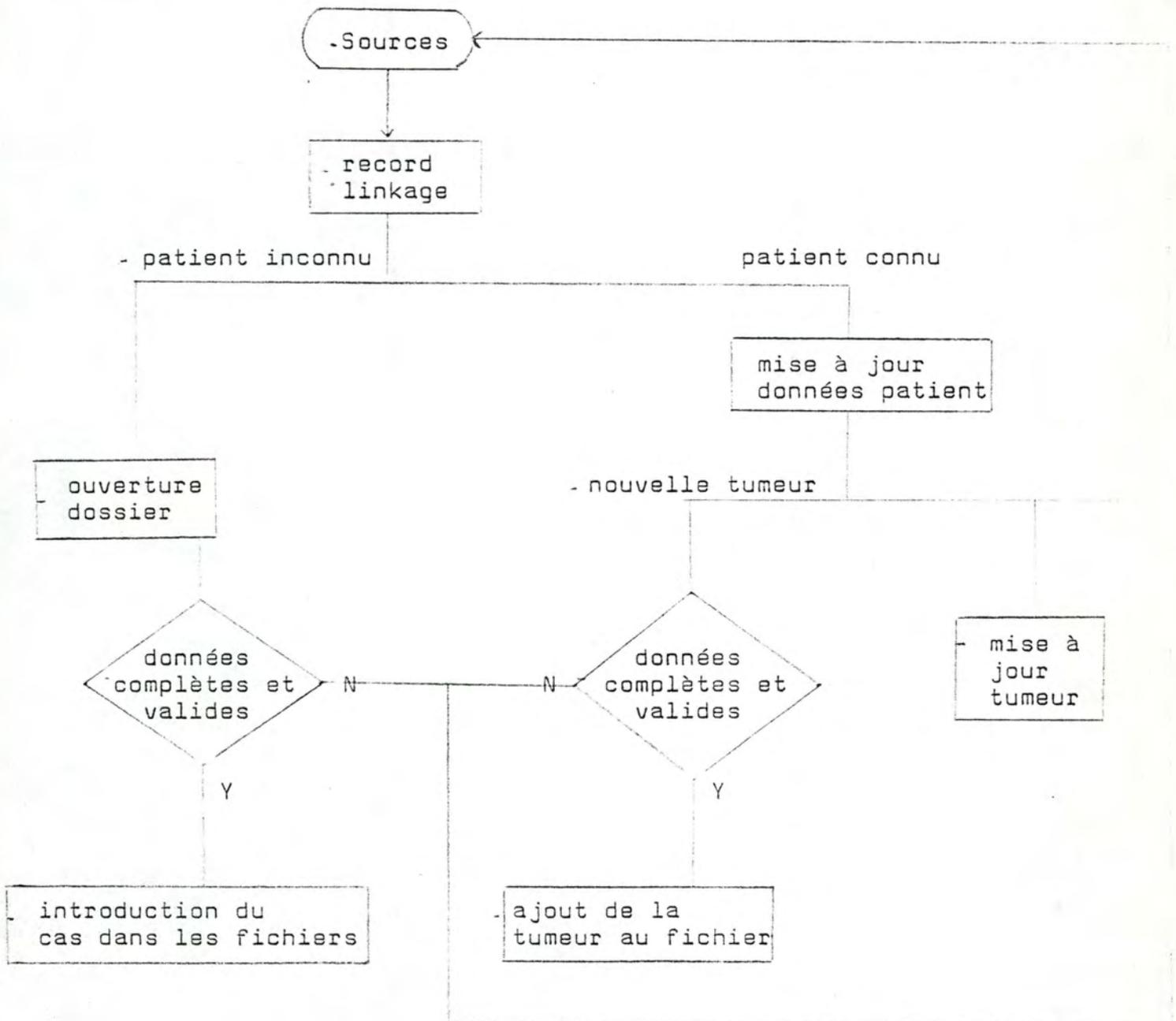
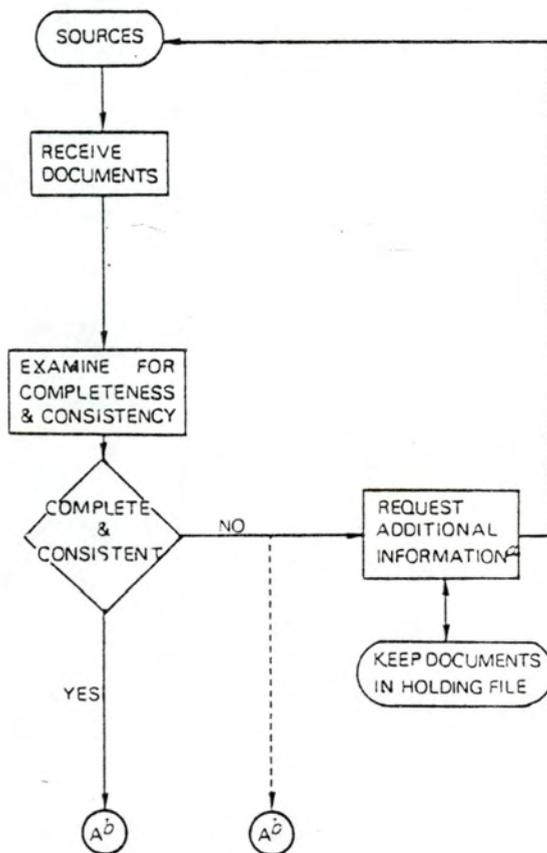


fig. III : Grandes étapes de l'input des données.

INPUT OPERATIONS

FIG. 7.2 ACCEPTANCE OF DOCUMENTS



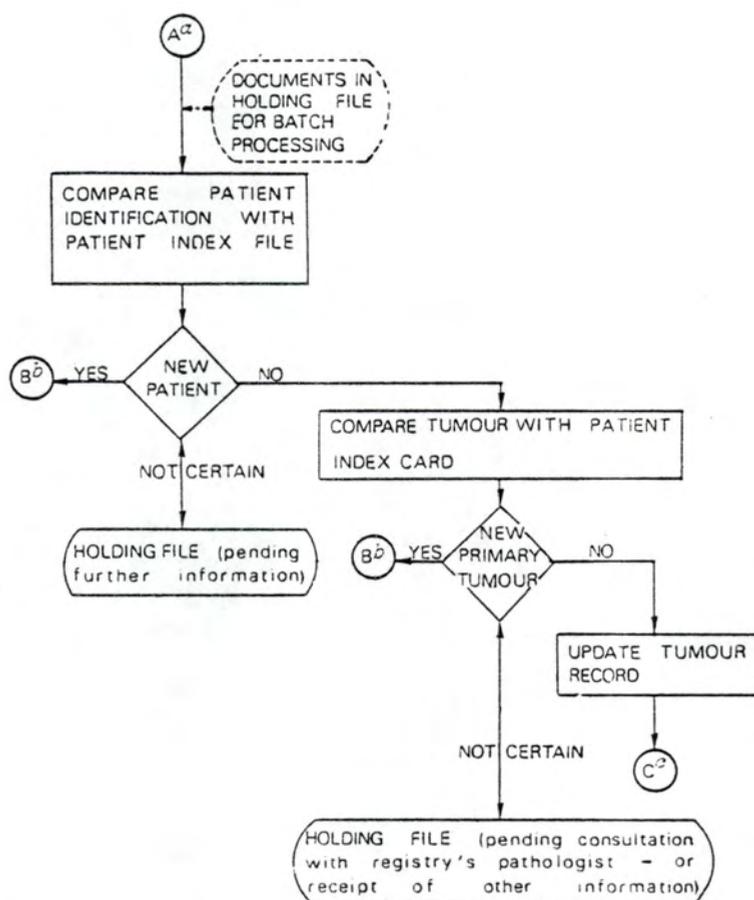
^a A source other than the original one may need to be contacted.

^b The next steps are shown in Figure 7.3.

fig.IVa

INPUT OPERATIONS

FIG. 7.3 MATCHING OF NEW MATERIAL TO IDENTIFY NEW OR OLD PATIENTS AND TUMOURS



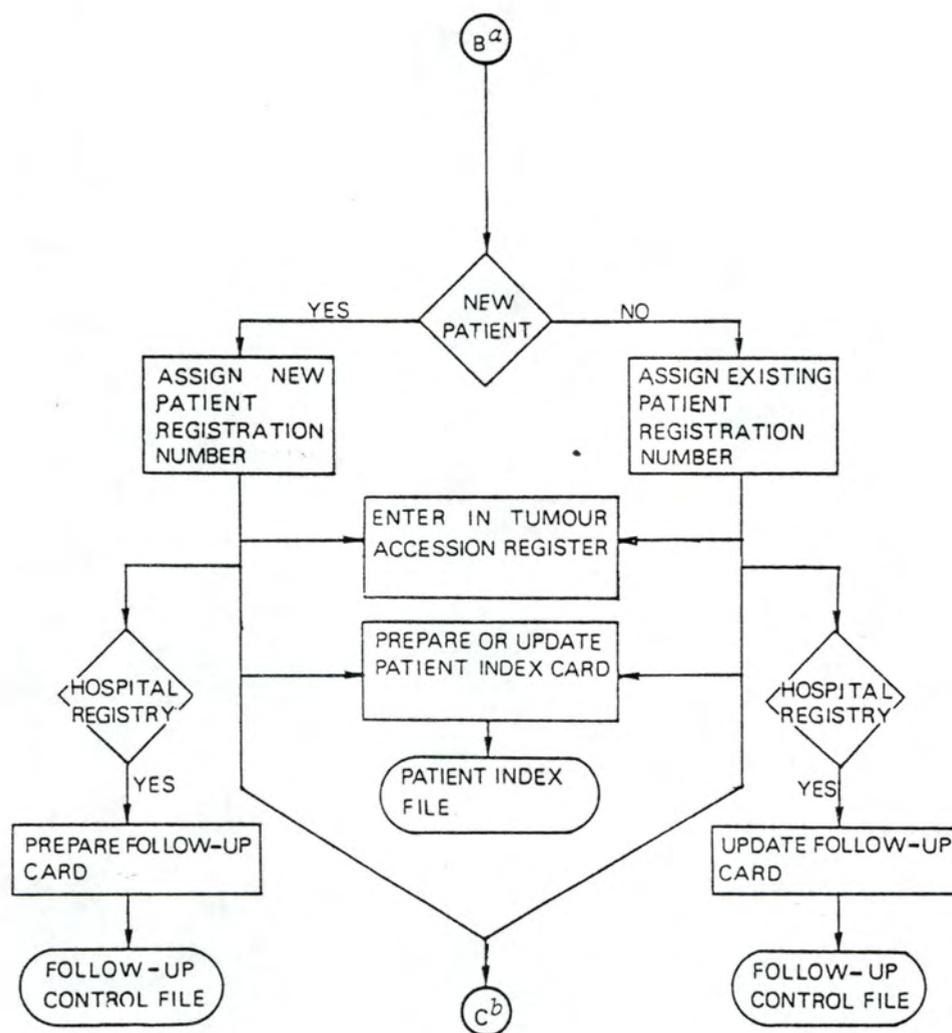
^a Continuation of Fig. 7.2

^b Continues in Fig. 7.5

^c Continues in Fig. 7.7

fig.Ivb

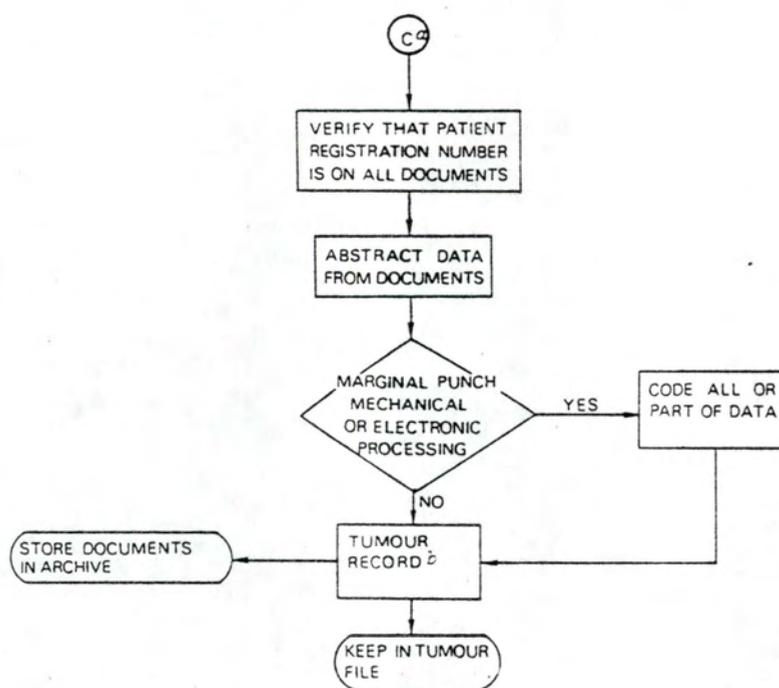
FIG. 7.5 REGISTRATION OF NEW TUMOURS



^a Continuation of Fig. 7.3

^b Continued in Fig. 7.7

FIG. 7.7 PREPARING AND UP-DATING THE TUMOUR RECORD



^a Continued from Figs 7.3 and 7.5

^b The tumour record may consist of an abstract card, coded cards, 80-column punch cards, computer tape, etc.

fig.Ivd

2.2.2 Autres fonctions.

2.2.2.1 Follow-up.

Dans le but d'établir des statistiques de survie, d'évaluer l'efficacité des traitements, de garder une base de données à jour..., les registres pratiquent un suivi des patients.

A cet effet, une ou deux fois par an, un formulaire est envoyé à la source d'information qui a fourni le cas. Ce formulaire reprend les renseignements principaux concernant le patient et sa tumeur et quelques questions concernant l'état actuel du malade.

Il s'agit essentiellement de savoir si la personne est toujours en vie et de connaître la date et la cause éventuelle de sa mort. On demande accessoirement de vérifier les données contenues dans le document.

La complémentation des informations (cfr. supra 2.2.1) se fait suivant le même principe. Les formulaires diffèrent suivant le type de renseignements demandés.

2.2.2.2 Edition de rapports.

Une collaboration étroite entre les sources d'informations et le registre conditionne de façon primordiale le bon fonctionnement de ce dernier.

Les sources alimentent le registre en données; le système ne doit pas fonctionner à sens unique.

Dès lors on assure l'édition annuelle d'un rapport reprenant des études statistiques de deux types concernant l'année écoulée.

Les résultats sont présentés sous forme de tableaux et d'histogrammes.

Le rapport consiste d'une part en des tabulations statistiques et d'autre part en des résultats d'études plus poussées (cfr. infra 2.2.2.3 calcul des taux).

Exemples :

- le tableau du nombre d'occurrences pour l'année 1979 des localisations par âge et par sexe;
- le tableau des occurrences pour les cinq dernières années des localisations par sexe tous âges confondus.

2.2.2.3 Etudes statistiques.

Classiquement on calcule les taux d'incidence, de mortalité, de survie... Ce travail se fait par rapport à une population de référence, la population de base du registre. Il peut aussi se faire par rapport à des populations standard (population mondiale, européenne, du pays dans lequel est implanté le registre). Dans ce cas ces taux sont dits standardisés.

Nous nous contenterons de définir brièvement ces concepts car ceci ne constitue pas l'objet substantiel de notre étude.

Pour la même raison, nous n'aborderons pas les autres fonctions spécifiques remplies par les registres, comme par exemple l'étude du cancer du sein chez la femme quadragénaire n'ayant jamais enfanté.

Notations :

Na = nombre de personnes années exposées à un risque donné dans une population donnée d'âge a

Ra = nombre de cas diagnostiqués dans la population considérée d'âge a (°)

Définition.

$$\text{Taux spécifiques} = Ta = \frac{Ra}{Na}$$

Standard usuels.

Les standard usuels sont la population mondiale, européenne, américaine (nord et sud)...

Il existe deux classes de standard :

soit que l'on considère la totalité de la population standard de référence;

soit que l'on considère une partie seulement de la population ("World truncated"); classiquement on considère les individus de 30 à 65 ans.

(°) Nous nous contenterons de définir les taux d'incidence standardisés ou non.

Taux standardisés.Définition.

Le taux standard est le taux que l'on observerait si la structure d'âge de la population était celle de la population standard.

I. Standardisation directe.a. par rapport à une population dite "standard" extérieure.

Soit W_a , le nombre de personnes années dans une population de référence d'âge a

Soit $W_1 \dots W_n$

$T_1 \dots T_n$ les taux spécifiques correspondants.

On définit le taux standardisé par rapport à la population extérieure par :

$$T_1 = \sum_a a W_a, \quad T'_1 = \sum_{a=a_0}^{a_i} R_a \frac{W_a}{N_a}$$

(taux tronqué)

b. Par rapport à une population interne.

Soit R_{ak} $a = 1 \dots A$
 N_{ak} $k = 1 \dots K$

qui ont la même définition que N_a et R_a ; l'indice k correspond aux sous-populations.

$$\text{Soit } W_a = \sum_{k=1}^K N_{ak} = N_a +$$

$$\text{on définit } T_{2k} = \sum_a \frac{R_{ak}}{N_{ak}} N_a +$$

le taux standardisé par rapport à la population interne k . ($k = 1 \dots K$)

II. Standardisation indirecte.

a. Par rapport à une population externe.

Soit toujours R_a et N_a définis ci-dessus
 et soit T_{a^*} le taux de référence standard donné
 pour l'âge a

On note le nombre de cas observés dans la population
 étudiée $O = \sum_a R_a$

Le nombre de cas attendus dans la population si elle
 était exposée aux taux standard

$$e = \sum_a T_{a^*} N_a$$

Le taux standardisé est défini par

$$T3 = \frac{O}{e} = \frac{\sum_a \frac{R_a}{a}}{\sum_a \frac{T a^*}{a} N_a}$$

b. Par rapport à une population interne.

Soit $R_{ak}, N_{ak} \quad k = 1 \dots N$

et

soit le taux standard $T a^* = \frac{\sum_k R_{ak}}{\sum_k N_{ak}}$

Le nombre de cas attendus sous le "taux général" est

$$\begin{aligned} e_k &= \sum_a T a^* N_{ak} = \sum_a \frac{\sum_k R_{ak}}{\sum_k N_{ak}} N_{ak} \\ &= \sum_a R_{a+} \frac{N_{ak}}{N_{a+}} \end{aligned}$$

l'observation

$$o_k = \sum_a R_{ak}$$

On définit

$$T4 = \frac{O_k}{e_k} = \frac{R_{+k}}{\sum_a R_{a+}} \frac{N_{ak}}{N_{a+}}$$

Les 4 taux standard définis ci-dessus sont utilisés.

Si on prend $T4$ il faut effectuer des tests d'homogénéité dans les courbes d'âges.

Toutes ces définitions mettent en évidence la nécessité de connaître avec précision les données démographiques concernant la population de base du registre.

2.2.2.4 Editions de listes.

Il s'avère très intéressant pour les gestionnaires des registres de pouvoir éditer des listes dont voici deux exemples types;

d'une part un relevé nominatif et alphabétique des patients, avec pour chacun d'eux : l'adresse, la date de naissance, le sexe, la localisation de la tumeur, ... Pareille structure est un moyen très simple de créer un fichier signalétique; et d'autre part une liste alphabétique et chronologique des cas par localisation de la tumeur pour faciliter des recoupements avec d'autres fichiers tels que des fichiers de mortalité. (°)

On peut, par exemple, éditer la liste des patients d'un médecin et la leur envoyer.

A priori, on ne distingue pas l'utilité d'une telle liste pour le médecin mais on a constaté qu'une pareille pratique était psychologiquement nécessaire : les médecins ne restent pas sur l'impression d'un système fonctionnant à sens unique (sources vers registre).

(°) Les registres ne disposent pas toujours de bandes magnétiques contenant ces fichiers, qui leur permettraient de réaliser une fusion.

2.2.2.5 Conclusion.

Nous présentons en illustration, dans l'annexe 1, les rapports succincts des visites que nous avons effectuées aux registres de Grenoble et Lausanne.

Toutes les fonctions décrites ci-dessus peuvent être automatisées, partiellement ou totalement.

Notre système prendra en charge ces fonctions mais pas toutes dans l'immédiat (cfr infra 2.3.2).

2.3 Les registres du cancer dans les pays en voie de développement.

2.3.1 Identification du patient.

Dans les pays occidentaux, chaque individu peut être identifié sans ambiguïté. Il a un nom, une date de naissance précise, un numéro de carte d'identité, de sécurité sociale... Néanmoins, reste le problème de l'orthographe des noms; on peut pallier cet inconvénient par les autres renseignements ou utiliser des techniques telles que le Soundex pour encoder les noms propres.

Mais dans les pays en voie de développement, un homme possède rarement une pièce d'identité et s'il en a une ce n'est pas nécessairement la sienne (celle de son frère mort quelques mois avant sa naissance, par exemple).

Le nom évolue avec l'âge et les coutumes.

La date de naissance n'est presque jamais connue avec précision, au mieux sait-on le mois et l'année. On peut éventuellement retrouver une approximation si la naissance est intervenue lors d'un événement marquant tel une fête tribale.

En conséquence, nous n'avons aucune donnée propre au patient commune à tous ces pays qui permette de l'identifier avec certitude.

Comment dans ces conditions réaliser un record linkage efficace ?

2.3.2 Statistiques.

Le calcul des taux d'incidence ... se fait par rapport à une population de référence qui est le plus souvent la population de base du registre. On ne dispose généralement pas des données démographiques nécessaires au calcul de ces taux. C'est pourquoi nous avons délaissé la prise en charge des calculs par notre système, mais seulement provisoirement (cfr. infra chap. 6 : prolongements).

2.3.3 Sécurité de la base de données.

Les pays en voie de développement disposent de moyens techniques précaires.

L'alimentation en énergie est intermittente. On doit s'attendre, à tout moment, à des coupures de courant.

Si de tels accidents se produisent pendant que l'on met à jour la base de données, celle-ci peut se trouver dans un état cohérent (modification inachevée).

Il faut donc prévoir une restauration de la base de données dans un état cohérent après de tels incidents.

Chapitre 3. Exposé de la solution.

3.0 Introduction.

Nous allons dans ce chapitre énoncer les principes de la solution que nous proposons.

Son implémentation sera développée ultérieurement.

Remémorons-nous les données du problème :

- le système est destiné à de petits registres (enregistrement de 2000 cas par an);
- ces registres disposent de peu de moyens techniques et financiers;
- le logiciel sera utilisé par du personnel non familiarisé avec l'informatique;
- il doit pouvoir faire face à des problèmes touchant la sécurité de la base de données (fréquentes pannes d'alimentation);
- nous devons résoudre un important problème d'identification d'un individu,
- le budget alloué est de 10.000 \$.

La fonction principale d'un registre du cancer est la collecte des données. C'est donc à cette fonction que nous avons consacré la plus grande partie de nos efforts. Nous allons étudier au paragraphe suivant le scénario de cette fonction.

Connaissant le peu de moyens des registres auxquels notre système est destiné, nous avons limité nos ambitions en ce qui concerne les autres fonctions.

Nous avons fait en sorte que l'édition de rapports et un follow-up réduit soient possibles.

Nous permettons aussi l'interrogation de la base de données tant à l'écran qu'à l'imprimante.

Nous avons, par contre, laissé tomber les statistiques telles que le calcul des taux standardisés ou non : les registres ne disposent pas des données démographiques nécessaires à ces calculs. Ce sera une extension possible à ce logiciel.

Caractéristiques générales :

Le problème d'identification d'un individu ne possède pas de solution unique, dans le sens qu'il n'existe pas de données communes à tous les pays qui permettent d'identifier un être humain : nous proposons dès lors une "identification à la carte" : nous proposons dès lors une solution paramétrable (cfr. infra 3.3).

La collecte des données se fera au travers de masque de saisie. On pourra interroger la base de données sous base de critères qui seront saisis à l'aide de ces mêmes masques.

Le système sera "truffé" de messages à l'intention de l'utilisateur. Lors de la saisie des données, via les masques, l'utilisateur pourra interroger le système pour savoir ce qu'il doit entrer : un texte d'aide sera associé à chaque zone du masque de saisie.

Enfin, des procédures de sécurité sont prévues pour le

cas échéant rétablir la base de données dans un état cohérent.

Le contenu des différents fichiers est arbitraire. Il reprend les données les plus fréquemment utilisées dans les registres que nous avons visités.

Nous allons développer les différents points évoqués ci-dessus. Nous cloturerons ce chapitre en présentant le matériel et le langage de programmation choisi.

Nous justifierons ce choix.

3.1 Input des données.

Nous savons que des données concernant un même patient peuvent provenir de différentes sources et qu'il faut dès lors recouper les informations entre elles et avec le contenu des fichiers existants; la phase cruciale est l'identification du patient (record linkage). Notre idée est d'interroger directement à l'écran la base de données : l'utilisateur entre quelques critères sur le patient et/ou sur sa tumeur.

Le système affichera alors un résumé de tous les cas répondant aux critères entrés. Si le cas traité par l'utilisateur est dans la liste, il sera sélectionné par ce dernier et éventuellement mis à jour.

Si le patient ne se trouve pas dans la liste, l'utilisateur pourra alors s'il le désire entrer le cas.

La saisie des données se fera au travers de deux masques de saisie, l'un destiné à recevoir les données-patient, l'autre les données tumeurs (ces données sont trop nombreuses que pour être saisie en un seul écran).

Un maximum de données est codifié suivant les codifications internationales proposées par l'O.M.S. (voir contenu des fichiers).

Les données sont validées par le système.

On vérifie la chronologie des différentes dates : date de naissance date du diagnostic date de la mort par exemple.

Les codifications sont validées (existence du code).

Différents contrôles croisés sont réalisés :
on vérifie la compatibilité entre le sexe du patient et la localisation de sa tumeur, entre le site de la tumeur et l'histologie, entre l'histologie et l'âge du patient (des types de cancer sont improbables chez les enfants de moins de 10 ans) et enfin entre le site de la tumeur et la latéralité.

Des avertissements seront donnés aux utilisateurs :
par exemple si l'histologie a été précisée, nécessairement il y a eu un examen de laboratoire; on demandera alors à l'utilisateur de préciser, si cela est possible, la source de la donnée.

Nous conseillons la tenue à jour d'un fichier signalétique des patients : il n'est pas possible d'introduire une recherche phonétique sur le nom des patients car cette recherche est dépendante de la langue du pays, de plus on connaît rarement l'orthographe d'un nom.

La recherche par l'oeil humain dans un fichier signalétique reste plus rapide; l'intelligence humaine peut faire des rapprochements qui sont impossibles à réaliser avec une machine, des facteurs subjectifs entrent en jeu, la sélection est beaucoup plus fine.

Bien évidemment, parallèlement au fichier informatique, on tiendra à jour des dossiers manuels :
ils contiendront tous les documents reçus au registre concernant le patient (un dossier par patient). Le dossier et tous les documents porteront un même numéro qui

sera d'ailleurs le numéro d'enregistrement dans le fichier informatique.

3.2 Les autres fonctions.

Interrogation de la base de données.

L'utilisateur pourra effectuer des sélections dans la base de données sur n'importe quels critères. Ceux-ci seront saisi à l'aide des masques utilisés pour l'input des données. Un résumé de tous les cas répondant aux critères sera affiché.

L'utilisateur peut sélectionner tout ou une partie de ces cas et les visualiser en détails : le cas total est affiché dans les masques de santé. L'utilisateur peut alors éventuellement modifier ces cas.

Les cas sélectionnés peuvent être édités à l'imprimante.

Follow-up.

Le follow-up tel que nous l'envisageons n'est pas automatique.

Quand l'utilisateur lancera la fonction de follow-up, tous les cas pour lesquels on n'a plus de nouvelles depuis au moins un an sont sélectionnés et une lettre de suivi sera éditée pour chacun de ces cas; cette lettre sera envoyée à la source d'informations; on demande dans cette lettre des données sur l'état actuel du patient.

Statistiques.

Les moyens des registres auxquels notre système est destiné sont limités. Ils ne disposent des données suffisantes à l'établissement de statistiques élaborées.

Nous nous contentons dans un premier temps de permettre des calculs de répartition par âge et par sexe, des localisations des tumeurs et la répartition par sexe et par année de référence (année de diagnostic) des localisations des tumeurs (évolution du nombre de cas au cours des ans pour évaluer l'efficacité de la prévention par exemple).

Ces deux tabulations statistiques sont suffisantes pour l'édition des rapports annuels (cfr. supra).

3.3 Paramétrisation du système.

3.3.1 Solution au problème d'identification.

Il n'existe pas de solution unique et psychologiquement acceptable au problème d'identification d'un individu. On pourrait, par exemple, passer à coder les empreintes digitales; une empreinte peut en effet être représentée par une trentaine de nombre. On peut identifier un être humain avec un taux d'erreur de 10^{-6} à l'aide de ces nombres, mais outre que cette solution est psychologiquement difficilement acceptable, elle nécessite de la part de la personne qui sera chargée de cet encodage un long apprentissage.

Dans les pays en voie de développement on utilise des informations fort différentes pour identifier un individu : son nom, son surnom, le nom de la tribu, le nom du sorcier du village, les noms des parents... Cela dépend des pays et des coutumes de ceux-ci.

Nous sommes dès lors contraints de proposer une solution paramétrable. Examinons-là :

l'enregistrement patient comportera une partie indentification du patient dont la longueur devra être inférieure à une valeur maximale donnée.

Elle comprend deux parties :

- la première est fixe dans le sens où la signification des variables qui la compose est donnée une fois pour toute; elle contient le nom, le nom de jeune fille, le sexe, et la date de naissance; la longueur maximale des noms

reste un paramètre de cette zone.

- la seconde partie est entièrement paramétrable; l'utilisateur décidera à l'initialisation du système ce qu'elle contiendra (*); au maximum elle peut contenir 8 variables.

((*) en fonction des données dont il dispose pour identifier un individu).

Les seules contraintes sont, pour rappel, le nombre de variable (8 maximum) et la longueur totale de la partie identification qui ne peut excéder une valeur maximale donnée.

Typiquement, dans nos pays, ce seraient les prénoms et l'adresse du patient, et un éventuel changement d'adresse.

3.3.2 Le principe de paramétrisation est étendu à l'ensemble du système.

Nous proposons un contenu arbitraire pour les fichiers (cfr. infra 3.4).

L'utilisateur, à l'initialisation du système, choisira les items qu'il décide de retenir; les données disponibles étant fonction des moyens, et pourra ainsi se construire un registre du cancer sur mesure.

Nous avons ajouté à l'enregistrement tumeur 6 variables sans signification. L'utilisateur choisira le sens à donner à ces variables, soit qu'il dispose d'une donnée supplémentaire, soit qu'il se lance dans une étude particulière nécessitant des données spécifiques.

3.4 Les fichiers.

3.4.1 La découpe en fichiers.

Nous avons choisi de mettre les informations concernant le patient et celles concernant sa(s) tumeur(s) dans des fichiers distincts :

Deux raisons justifient ce choix :

- les statistiques ont pour objet les tumeurs et non pas les patients, les données sur le patient n'étant enregistrées que pour éviter des enregistrements doubles;
- un même patient peut avoir plusieurs tumeurs. Si on enregistrerait toutes les données (patient et tumeur) dans un même fichier, nous aurions des informations redondantes puisque nous avons un enregistrement par tumeur et que les données concernant le patient se retrouveraient dans chaque enregistrement. Cela n'est pas trop grave car ces cas sont rares et par conséquent la perte de mémoire de masse serait négligeable.
Mais on désire éviter des redondances de traitement : si l'on doit, par exemple remettre à jour l'adresse du patient, on doit accéder à tous les enregistrements de ce patient et mettre la donnée à jour partout.

Nous avons donc un fichier-patient et un fichier-tumeur.

3.4.2 Contenu du fichier.

3.4.2.0 Introduction.

Le choix du contenu des fichiers résulte des considérations suivantes :

- la plupart des registres que nous avons visité, enregistrent énormément de données ou tout au moins ont prévu de le faire : les données sont trop souvent très spécifiques et sont rarement disponibles. Une partie des données enregistrées n'est pas utilisée sinon pour des études plus particulières; mais, dans ce cas on doit mettre à jour ces données souvent très anciennes. Pourquoi alors ne pas les collecter au moment où on en a besoin ?

C'est un luxe que ces registres ne peuvent se permettre : ils travaillent souvent en étroite collaboration avec un centre universitaire ou hospitalier qui dispose de grands moyens techniques et le gaspillage de place mémoire n'est pas un problème.

- n'oublions pas que le système est destiné aux pays en voie de développement et que les registres dans ces régions ont essentiellement comme fonction la saisie des données. Les études statistiques, et l'édition de rapports annuels sont confiés à des centres plus importants.

Nous avons décidé, en commun accord, avec les épidémiologistes du C.I.R.C., de ne conserver que les données réellement utiles en laissant toutefois la possibilité au responsable du **registre** d'enregistrer quelques autres informations dans des variables auxquelles nous n'avons donné aucune sémantique (cfr. supra 3.3.2 et infra 3.4.2.2).

3.4.2.1 Fichier-patient.

Nous allons examiner un à un les items du fichier et voir quelle est leur utilité. Entre parenthèses, la longueur maximale de chaque item est précisée.

numero enregistrement patient (6)

Dès qu'un nouveau cas est enregistré, le système lui attribue automatiquement un numéro qui sera aussi son numéro de dossier. L'utilisateur n'a accès à cet item qu'en lecture. Tous les documents reçus au registre concernant ce patient porteront ce numéro. Nous verrons ultérieurement que sur base de ce numéro, nous pourrions accéder aux tumeurs de ce patient.

identification du patient

zone fixe

nom du patient

Il s'agit du nom USUEL du patient. Rappelons que dans les pays en voie de développement, un individu peut avoir plusieurs noms.

nom de jeune fille et sexe du patient (1)

Ces items ne nécessitent pas d'explication particulière. La donnée sur le sexe est codifiée.

date de naissance (7)

Cette donnée ne peut être enregistrée que si l'on est absolument certain de son exactitude. Elle peut être incomplète : seulement le mois et ou l'année.

âge (2)

Si on ne connaît pas la date de naissance avec exactitude, on enregistre une estimation de l'âge à la date d'enregistrement. Il faut donc aussi enregistrer la date d'enregistrement.

lieu de naissance (5)

Cette information est codifiée (code postal par exemple).

zone à la carte

Elle comprend 8 items au maximum (ident1 à ident8). Les données peuvent être codifiées ou non.

profession (2)

La donnée est codifiée suivant la classification internationale des professions (Genève).

ethnie et état-civil (1)

Ces données sont aussi codifiées.

Les 3 derniers items servent à déterminer des facteurs et populations à risque. Ainsi, on constate plus de cancers du sein chez les femmes n'ayant jamais enfanté que chez les autres.

date du dernier contact (7)

Cette donnée est utilisée par la fonction de follow-up. Si on n'a pas reçu de nouvelle depuis un an concernant un patient, des informations sur son état actuel sont demandées aux sources d'information qui ont fourni le cas.

date et cause éventuelle de la mort (7 et 4)

On établit à l'aide de ces données des statistiques de

survie et de mortalité. La cause de la mort est codifiée suivant une codification internationale.

degré de complétude (1)

Tous les cas, à condition que l'on puisse identifier le patient, sont enregistrés. Rappelons-nous le fonctionnement général des registres : les cas incomplets étaient mis en attente dans un fichier spécial en attendant les informations complémentaires.

Nous décidons de ne prendre qu'un seul fichier et d'ajouter à chaque enregistrement un indicatif de complétude qui servira à retenir ou non les cas lors des calculs statistiques.

nombre de tumeurs (1)

Cet item nous permettra, avec le numéro d'enregistrement, d'accéder à toutes les tumeurs du patient (cfr. infra 3.4.2.2).

3.4.2.2 Fichier-tumeur.

N° enregistrement tumeur (6)

Il est, comme le numéro d'enregistrement patient, attribué automatiquement par le système.

Il est le résultat de la concatenation du numéro enregistrement du patient et de l'occurrence de la tumeur pour ce patient. A partir du numéro patient et du nombre de tumeurs on peut donc accéder à toutes les tumeurs du patient.

date de référence (7)

C'est souvent la date du premier diagnostic ou des premiers examens de laboratoires.

Cette date est utilisée pour des calculs de survie.

Site ou topographie (4)

C'est la codification de la localisation de la tumeur suivant le code ICDO (International codification of diseases-Oncology).

Quelques données médicales *

Histologie ou morphologie (5)

Code ICDO - résultat de l'analyse en laboratoire des tissus.

Le stade (4)

code TNM

et

L'étendue (1)

de la tumeur sont précisés

Latéralité (1)

(gauche - droit - central - sans objet ...)

Sources des informations (5 x 4)

On permet d'enregistrer 5 sources différentes.

Rappelons que l'on peut recevoir des données de plusieurs sources pour un même cas.

Ces sources sont codifiées.

Ces données sont utiles pour réaliser le follow-up et la complémentation des informations.

Bases du diagnostic (11 x 1)

On précise les bases du diagnostic avec parfois le degré de certitude : on propose une liste de bases de diagnostic possibles; on donne comme valeur à ces items 0 ou 1 selon que ce soit ou non une base de diagnostic. Parfois, comme à Lausanne, on y met le degré de certitude du diagnostic.

Les bases possibles retenues sont :

- la radiographie
- la cytologie
- l'autopsie
- la biopsie
- la chirurgie
- le diagnostic purement clinique
- la biochimie
- l'immunologie
- l'hématologie
- inconnu
- autre.

De la même manière, on enregistre les traitements appliqués (7 x 1)

Les traitements possibles retenus sont :

- la radiothérapie
- la chimiothérapie
- la chirurgie
- l'hormonothérapie
- autre
- inconnu
- rien (aucun traitement)

6 items' libres' (6 x 5)

On applique le même principe que pour la zone d'identification dans le fichier patient.

L'utilisateur décidera lui-même ce qu'il désire ranger dans ces variables.

degré de complétude (1)

idem fichier patient.

3.4.2.3 Taille des fichiers.

Rappelons que le système doit enregistrer deux mille cas par an pendant cinq ans.

un enregistrement-patient	=	255 bytes
un enregistrement-tumeur	=	97 bytes
		<hr/>
		352 bytes

2000 enregistrements pendant 5 ans \simeq 3,5 Megabytes.

A cela doivent s'ajouter les programmes et les dictionnaires de code 0,5 Megab. et une marge de sécurité.

Ces chiffres justifient l'achat d'un disque dur de 5 Megab (cfr. infra 3.6).

3.5 Validations.

Mis à part les classiques contrôles de bornes et de validité des dates, des contrôles croisés doivent être prévus.

Il faut vérifier la compatibilité entre :

le site et l'histologie

le site et le sexe

l'histologie et l'âge du patient.

Il faut en outre vérifier que la chronologie des dates est respectée :

date de naissance date de référence

date du dernier contact date de la mort.

Toutes ces validations ne sont pas nécessairement impératives :

ainsi le contrôle croisé sur l'histologie et l'âge : certaines valeurs sont improbables pour un enfant âgé de moins de 10 ans; on demandera à l'utilisateur de confirmer les données, car celles-ci étant tellement improbables que l'on est en droit de croire à l'erreur.

Un autre exemple : si une valeur a été attribuée à l'histologie, nécessairement le patient a subi des examens de laboratoires. Si aucune source de ce type n'est mentionnée, il faut le signaler à l'utilisateur.

3.6 Choix du matériel.

Le logiciel développé doit répondre au critère de portabilité; c'est pourquoi nous avons opté pour un système tournant sous CP/M; la majorité des microordinateurs tournant sous ce système d'exploitation (°).

L'importance des programmes à mettre en oeuvre, par exemple doivent être présents en mémoire en même temps : CP/M; la gestion de base de données, la gestion d'écran, le programme de saisie des données avec toutes les validations, nous laisse supposer que 128k seront nécessaires si nous voulons éviter des chainages de programmes.

La quantité d'informations à enregistrer est importante; un disque dur s'avère nécessaire.

Après une étude détaillée du marché, nous constatons que les matériels répondant à ces critères sont annoncés sur le marché mais ne sont pas encore disponibles (1).

Notre choix initial s'était porté sur le RAINBOW 100 de Digital. Ce matériel répond aux critères énoncés ci-dessus. Le RAINBOW est un biprocesseur (8 et 16 bits).

Le système possède un programme d'autodiagnostic qui peut s'avérer très utile si on se rappelle les

(°) Un tel système est bâti autour d'un processeur compatible 8080.

(1) Des matériels répondant à ces critères existaient déjà mais pas pour 10.000 \$; budget qui nous était alloué.

conditions de fonctionnement du système (éloigné de toute assistance technique). *

A l'époque le RAINBOW n'était pas disponible en Europe, et de toute façon les logiciels pour les 16 bits n'étaient pas encore tout à fait au point et notamment CP/M 86.

Cette dernière remarque nous a conduit à éliminer des systèmes 16 bits tel que le SIRIUS.

Un autre choix possible aurait été le ZENITH 100, mais il n'était pas encore disponible avec disque dur.

Notre choix s'est finalement porté sur la 'boîte noire' de RAIR (modèle 3130). Pour plus de détail sur ce matériel, consulter la fiche technique à la page suivante.

Ce matériel qui est un 64k est suffisant pour développer le logiciel. Il n'est pas impossible qu'une deuxième version du logiciel soit disponible sur un 128k dans les tous prochains mois.

* Il est évident que la réputation de Digital n'était pas étrangère à ce choix.

Spécifications de la Boîte Noire 3/30.

Micro-Processeur

Micro-processeur	8085A
Horloge temps réel	2 x fréquence de ligne
Interruptions	4 niveaux
DMA (accès direct mémoire)	4 canaux mode continu
Gestion de la mémoire	Topographie de l'adresse des pages 16 x 8 bits
Option processeur arithmétique	Opérations en virgule fixe et flottante 16 et 32 bits

Mémoire RAM (Mémoire à accès sélectif)

Capacité de base de la mémoire	64 ko
Capacité maximale de la mémoire	512 ko
Incréments d'extension	64 ko
Caractéristiques de la mémoire	Régénération transparente, sans états d'attente

Points de connexion entrée/sortie en série

Type d'interface	RS-232C/CCITT V24 compatible
Vitesse d'exploitation	de 50 à 19.200 bauds
Modes de communication	synchrones/asynchrones
Nombre maximal d'interfaces	16

Disque fixe intégral

Type d'entraînement	Micro-Winchester 5,25"
Mode d'enregistrement	MFM
Capacité	6.380 ko brut
Vitesse de transfert	5.000.000 bits/sec.
Temps d'accès	3ms piste à piste

Floppy disque

Type d'entraînement	Disquette 5,25" à double tête
Mode d'enregistrement	MFM (double densité)
Format d'enregistrement	Implantation des secteurs fixée par programme
Capacité	500 ko brut
Vitesse de transfert	250.000 bits/sec.
Temps d'accès	5ms piste à piste

Option de bus parallèle

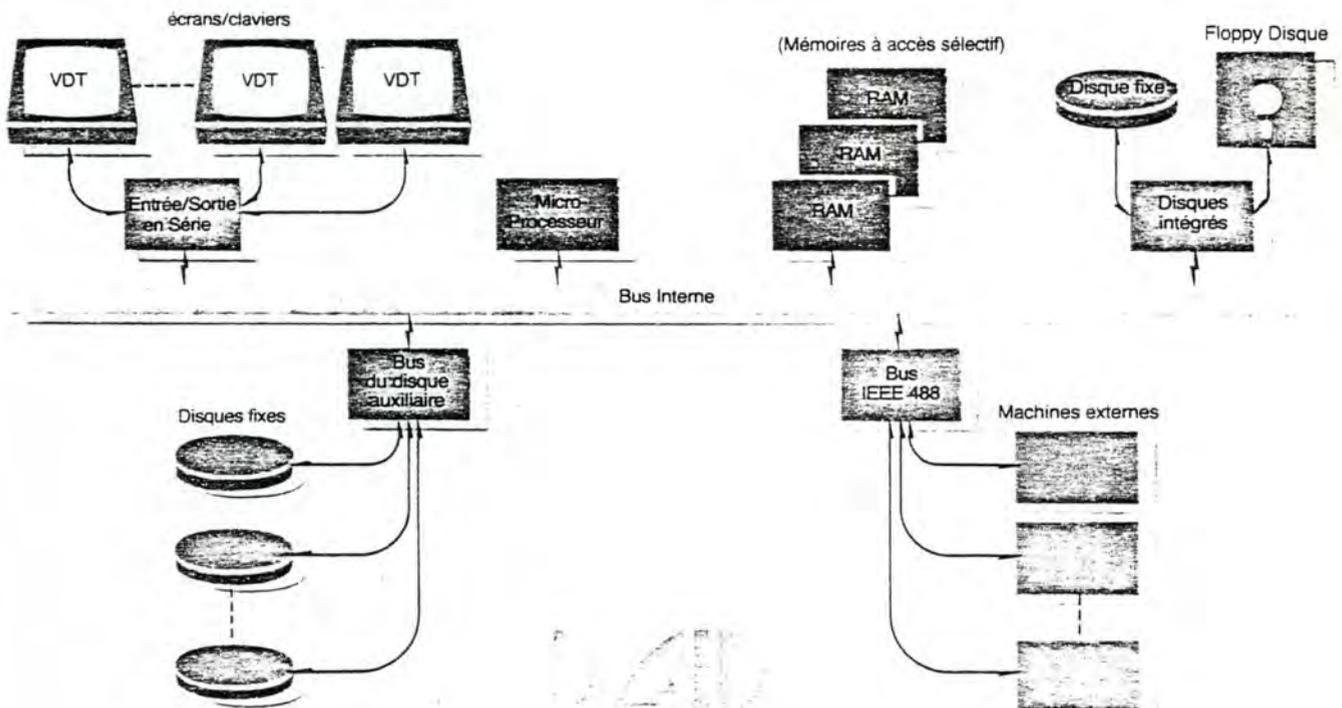
Type d'interface	IEEE 488 standard
Vitesse d'exploitation	800 ko/sec.
Modes d'exploitation	contrôleur/émetteur/écouteur
Longueur maximale de bus	20 mètres

Option de disque auxiliaire

Type d'entraînement	8" Winchester
Capacité d'entraînement	11.500 ko brut
Nombre maximal d'unités	8

Général

Alimentation à découpage	115/230 Vac \pm 10%, 50/60 Hz 3,15 A max
Dimensions: Largeur	508 mm
Profondeur	406 mm
Hauteur	152 mm
Ambiance d'utilisation	de 10° à 40° C Humidité relative: de 20 à 80 %
Poids	15,9 kg



France

11 bis rue du Colisée - 75008 Paris - Tél. 256.10.78 - Télex 290 177 RAIR SA

United Kingdom

6-9 Upper St Martins Lane - London WC2H 9EQ - Tel. (01) 836.6921 - Telex 298452

West Germany

Clemensstrasse 5-7 - 5000 Köln - Tel. (0221) 219811 - Telex 8881915

USA

4101 Burton Drive Santa Clara - California 95050 - Tel. (408) 988 1790 - Telex 677038

3.7 Choix du langage.

Les responsables du projet au C.I.R.C. souhaitent que des non-initiés à l'informatique puissent modifier ou ajouter des modules au système, en vue d'études statistiques particulières ou encore pour ajouter au programme de validation quelques tests inhérents au choix des items.

Souvenons-nous aussi que le logiciel doit être portable.

Le MBasic de Microsoft répond parfaitement à ces deux objectifs.

Néanmoins, le premier objectif nous paraît relever un peu de l'utopie : un utilisateur qui désirerait modifier les programmes devrait être au fait du système d'exploitation CP/M, de la structure des programmes et de l'organisation de la base de données !

Pourquoi n'être pas parti d'un S.G.B.D. existant ?

Le seul S.G.D.B. disponible dans des délais raisonnables, compatible CP/M et RAIR était dbase II.

DBASE II ne permet pas l'utilisation de tableaux, or nous devons réaliser des tabulations statistiques.

Des systèmes de sécurité ne sont pas prévus. Dbase II ne permet pas l'appel à des modèles écrits en un autre langage : on peut sortir de dbase II, faire appel à une autre fonction puis entrer à nouveau dans dbase II au point où l'on était au moment de l'appel mais sans pas-

sage de paramètre. Il n'est donc pas possible d'y ajouter des modules extérieurs de sécurité.

Chapitre 4 : Sous-système développé.

4.0 Introduction.

Nous allons présenter longuement les deux sous-systèmes développés à savoir le système de gestion de base de données et la gestion d'écran.

Ces systèmes vont nous permettre de réaliser très facilement la paramétrisation du système : les descriptions des fichiers et des masques de saisie seront externes aux programmes de mise à jour de la base de données. Il suffit de tester, dans ces différents programmes, si la valeur est vide (°) avant de procéder au traitement de la donnée; si c'est le cas, la donnée ne sera pas traitée.

Il nous est impossible de donner ici les algorithmes détaillée pour l'implémentation des différentes fonctions par ces deux sous-systèmes; nous nous contenterons de donner une bonne idée de l'implémentation, en utilisant, si nécessaire, des exemples.

(°) Un item peut avoir une valeur vide parce que l'utilisateur n'a pas fourni de valeur pour cet item ou parce que cet item n'a pas été retenu à la paramétrisation du système.

4.1 Gestion de la base de données.

4.1.0. Introduction

Dans un contexte mono utilisateur , les programmes de gestion de base de données que nous avons développés sont destinés à pallier à des faiblesses du langage de programmation utilisé en l'occurrence le basic et à nous donner un outil rudimentaire de description , d'exploitation et de sauvegarde de la base de données.

Ce logiciel fait appel à quelques concepts du modèles relationnels et à une terminologie particulière.

Après un exposé des différentes primitives envisagées , nous présenterons une description formelle et technique. La description formelle est destinée à un programmeur d'application tandis que la description technique a pour objet de fournir un document très précis en vue de permettre des modifications éventuelles des algorithmes sous-jacent à chaque primitives.

4.1.1 Concepts de base.

4.1.1.0 Concepts de B.D relationnel.

Expose succinct du modele

Les composants de base.

.

L'information se presente sous la forme :

- 1> d'ensembles de valeurs de meme type appeles DOMAINES. Chaque VALEUR designe un element autonome du monde reel (domaine-entite) ou l'etat d'une propriete du monde reel (domaine-propriete),
- 2> d'ensembles d'associations entre valeurs de domaines , appeles RELATIONS. Les associations , appelees LIGNES ou N-UPLETS, d'une meme relation ont la meme structure. Une ligne de relation represente une association entre les constituants du monde reel designes par ces valeurs. Un domaine est utilise dans une relation en tant que ATTRIBUT de celle-ci. L'ensemble des lignes d'une relation peut etre representee sous la forme d'un tableau dont les colonnes correspondent aux attributs,
- 3> de diverses contraintes auxquelles sont soumis domaines et relations. Ces contraintes representent des proprietes du monde reel.

Une base de donnees conceptuelle est constituee :

- 1> d'un schema : texte decrivant les proprietes generales permanentes des domaines (nom, type de valeurs,..) et des relations (nom, liste des attributs, ...) ainsi que les contraintes.

2> des ensembles de valeurs des domaines et des ensembles de lignes des relations ; ces ensembles, contrairement au schéma , peuvent évoluer au cours du temps.

Sémantique.

.

A chaque domaine et chaque relation est associée une expression sémantique qui donne l'interprétation ou signification par rapport au monde réel.

LES DOMAINES.

Soit un ensemble d'information de même nature (désigne un élément du monde réel ex: la taille , le sexe, le nom ..). Nous conviendrons d'appeler DOMAINE un ensemble d'informations et VALEUR un élément de cet ensemble. Pour certains domaines nous pouvons préciser quels en sont les valeurs, soit en en donnant la liste :

$$\text{SEXE} = \{ \text{"masculin"}, \text{"feminin"} \}$$

soit en en donnant des propriétés :

$$\text{NOMPATIENT} = \{ \text{chaines de moins de 20 caractères} \}$$

LES RELATIONS.

Le monde réel est constitué d'éléments et propriétés en association mutuelle. Nous représenterons une association entre les éléments par une association entre les informations (valeurs) désignant ces éléments. Ainsi si "martin jacques" désigne un nom et "masculin" le sexe; le fait que le nom soit celui d'un patient de sexe masculin se représentera par un couple de valeurs.

("martin jacques","masculin").

Nous appellerons RELATION l'ensemble de toutes les associations de meme nature et nous appellerons LIGNE (ou n-uplet) de cette relation une telle association.

Nous donnerons a une relation un nom qui l'identifie. Le format general d'une relation est :

Nom de la relation (liste des domaines de la relation).

Exemple : PATIENT(NOM,SEXE).

Quant a l'ensemble des lignes a un instant determine, nous l'appellerons , l'EXTENSION de cette relation :

NOM	SEXE
martin jacques	masculin
walran pierre	masculin
folon dominique	feminin

← ligne } extension
 domaine

DEGRE D'UNE RELATION.

On appelle degre d'une relation le nombre de domaines sur lesquels elle est definie.

EX: le degre de la relation PATIENT(NOM,SEXE,AGE) est 3.

D'une maniere generale une relation de degre n est dite n-aire.

Toutes les lignes d'une relation n-aire contiennent exactement n valeurs prises chacune dans un domaine de la relation.

DOMAINES-FUISSANCES

Chaque valeur d'un tel domaine est elle-meme un ensemble de valeurs. Ex: PERSONNE(NOM.PRENOM*) dont l'extension serait :

MARTIN jules ,georges,paul

WARLAN pierre,marc

DOMAINES-RELATIONS

Un tel domaine est une relation, une de ses valeurs est un n-uplet. Exemples.

PATIENT(NOM,LOCALITE,CODEICD)

CODEICD(NUMERO,LIBELLE)

Un triplet de PATIENT serait :

(MARTIN,30 rue de la meuse liege,(141, tumeur sein)

4.1.1.1 Terminologie utilisee.

CLE DE RECHERCHE et NUMERO D'APPARTENANCE

On appellera cle de recherche d'une relation, une liste d'attributs. Ex: dans la relation PATIENT(NOM,SEXE,ADRESSE) NOM et SEXE pourraient etre declare comme appartenant a une meme cle de recherche.

De facon a les distinguer, nous les numeroterons et nous appellerons numero d'appartenance une valeur permettant de definir l'appartenance d'un attribut a une cle de recherche.

Ex : soit la relation PATIENT(NOM,SEXE,ADRESSE)

Si NOM et SEXE appartiennent a la premiere cle de recherche alors il possederons un meme numero d'appartenance soit 1.

Un attribut peut participer a plusieurs cles de recherches.

Ex: PATIENT(SEXE,NOM,ADRESSE).

cle de recherche numero 1 : SEXE, NOM

cle de recherche numero 2 : NOM , ADRESSE

liste des numeros d'appartenance de NOM : 1,2

liste des numeros d'appartenance de SEXE: 1

liste des numeros d'appartenance de ADRESSE : 2

TYPE DE RELATION

- - - - -

Le type d'une relation reflète le choix (fait par l'utilisateur au moment de la description des relations) de la representation physique des lignes de l'extension d'une relation et defini implicitement les possibilites d'accès aux informations.

Nous distinguerons quatre types de relation ; le type SEquentiel (SEQ), le type RELatif (REL), le type InDexe-Relatif (IDR), et le type InDexe-sequentiel (IDX).

- IDX : cle de recherche multiple, ajout en sequence des lignes a l'extension de la relation et possibilite d'effectuer des recherches via des cles de recherches.

- IDR : cle de recherche multiple, l'utilisateur doit assigner un numero identifiant a chaque ligne a ajouter a l'extension de la relation , il peut egalement se servir de ce numero pour lire cette ligne tout en gardant

la possibilite d'effectuer des recherches via des cles de recherche.

- REL : pas de cle de recherche, un numero identifiant doit etre attribue a chaque ligne ajoutee a la relation et doit etre rappele lors de la lecture et de la mise a jour de toutes lignes de l'extension de la relation.

- SEQ : pas de cle de recherche, creation & lecture sequentielles des lignes de l'extension de la relation.

4.1.2 Caracteristiques du S.G.B.D.

Il permet de decrire des relations definies sur des domaines simples (pas de domaines-puissances, ni de domaines-relations) et de definir au maximum cinq cles de recherches non necessairement distinctes. Il fournit un certains nombres de primitives permettant :

- . d'accéder aux lignes de l'extension d'une relation selon un certain ordre,
- . d'enlever une ligne a l'extension d'une relation,
- . de garantir une certaine securite de la base de donnees.

4.1.3. Description informelle des primitives.

4.1.3.0 Introduction.

La description informelle a pour but de donner une idee approximative des primitives developpees au cours du memoire.

Nous avons defini trois grandes classes de primitives.

. Les PRIMITIVES DE DESCRIPTIONS de la base de donnees. Elles permettent a un programmeur de construire , de modifier , de sauvegarder et restaurer la definition de la structure d'une base de donnees.

. Les PRIMITIVES D'EXPLOITATIONS de la B.D. donnent aux programmeurs la possibilite de modifier le contenu de la base de donnees.

. Les PRIMITIVES DE SECURITE permettent aux programmeurs de definir des operations "atomiques".

4.1.3.1 Les primitives de description de la B.D.

Nom : CREACAT

Parametres : Nom donne a la base de donnees , Nombre maximum de relations envisagees.

Effet : Memorise le nom d'une nouvelle base de donnees.

Nom : AJTREL

Parametres : Nom de relation , Degre maximum de cette relation , Type de relation.

Effet : Ajoute la description d'une relation a la description existante.

Restrictions : Il ne peut y avoir deux relations de meme nom dans une meme base de donnees et le nombre de relations existants doit etre inferieur au nombre maximum de relations communique par la primitive CREACAT.

Nom : AJTDOM

Parametres : Nom de relation , nom de domaine , longueur maximale que peut atteindre la valeur de ce domaine , numero d'appartenance (voir Terminologie particuliere).

Effet : Declare qu'un domaine d'un certain nom est defini pour une certaine relation.

Restrictions : Il ne peut y avoir deux domaines de meme nom pour une meme relation.

Nom : CHGNOM

Parametres : nom de relation , nom de domaine ,
nouveau nom de domaine , nouveau numero d'appartenance.

Effet : Change le numero d'appartenance, et le nom
d'un domaine d'une certaine relation.

Restrictions : Identiques aux precedentes.

Nom : SAVEBD

Parametre : nom de la base de donnees.

Effet : Sauve la description de la base de donnees
sur un fichier particulier.

Restrictions : Il ne peut y avoir deux bases de
donnees de meme nom.

4.1.3.2 Les primitives d'exploitation de la B.D.

Primitives d'accès a un domaine.

Nom : READOM.

Parametres : Nom de relation , nom de domaine.

Effet : donne la valeur courante d'un domaine d'une
ligne d'une relation dernièrement lue ou initialisee par
certaines primitives d'accès a la Base de donnees ou par la
primitive MAJDOM.

Nom : MAJDOM.

Parametres : Nom de relation , nom de domaine ,
chaine de caracteres.

Effet : initialise la valeur d'un domaine d'une future ligne d'une relation avec une chaine de caracteres.

Primitives d'accès à la base de données.

- - - - -

Nom : LOAEDB

Parametre : Nom de la base de donnée.

Effet : Charge en memoire la description d'une base de données. Cette B.D. est designee par un nom qui l'identifie parmi l'ensemble des bases de données disponibles.

Nom : OPEN

Parametres : Nom d'une relation, mode d'ouverture desire.

Effet : Effectue l'ouverture du fichier correspondant a une certaine relation selon le mode desire.

Nom : CLOSE

Parametre : Nom d'une relation.

Effet : Effectue la fermeture du fichier correspondant au nom de la relation donne.

Nom : START

Parametres : nom d'une relation, numero de cle de recherche.

Effet : recherche parmi les lignes de l'extension

d'une relation la premiere ligne dont la valeur de la cle de recherche est directement superieure ou egale a la valeur de la cle d'accès.

Restriction : Cette primitive n'est utilisable que si le mode d'ouverture choisi a l'OPEN est la lecture et que le type de relation est IDX ou IDR.

Nom : READNEXT

Parametre : nom d'une relation.

Effet : acces a la ligne dont la cle de recherche est directement superieure a la derniere cle de recherche obtenue par un START ou un READNEXT sur cette relation.

Restriction : Cette primitive n'est utilisable que si le mode d'ouverture choisi a l'OPEN est la lecture et que le type de relation est IDX ou IDR.

Nom : DELETE

parametres : nom d'une relation, (numero identifiant).

Effet : efface logiquement une ligne d'une relation.

si le type de la relation est :

- IDX on efface la derniere ligne lue par un start ou readnext;

- IDR & REL on efface la ligne dont le numero est mentionne.

- SEQ on efface la derniere ligne lue par un READ.

Restriction : Cette primitive n'est utilisable que si

le mode d'ouverture choisi a l'OPEN est la lecture.

Nom : READ

Parametres : nom d'une relation, (numero identifiant).

Effet : Effectue la lecture d'une ligne d'une relation.

si le type de la relation est :

- IDR & REL on lit la ligne dont le numero identifiant est mentionne;

- SEQ Lecture sequentielle (dans l'ordre de l'ajout des lignes a la relation.);

Restrictions : Cette primitive n'est pas utilisable que si le mode d'ouverture choisi a l'OPEN est l'écriture.

Nom : REWRITE

parametres : Nom d'une relation, (numero identifiant)

Effet : Reecrire une ligne lue et eventuellement modifiee.

si le type de la relation est :

- IDX & IDR Reecriture de la derniere ligne lue par un START ou READNEXT

- REL & SEQ Reecriture de la derniere ligne lue par un READ.

Restriction : Cette primitive n'est utilisable que si le mode d'ouverture choisi a l'OPEN est la lecture.

Nom : WRITE

Parametres : Nom d'une relation, (numero identifiant)

Effet : Effectue l'écriture d'une nouvelle ligne
d'une relation.

si le type de la relation est :

- IDX & SEQ Ecriture sequentielle.;

- IDR & REL On ecrit une nouvelle ligne

dont le numero identifiant est mentionne.

Restriction : Cette primitive n'est utilisable que si
le mode d'ouverture choisi a l'OPEN est l'ouverture.

4.1.3.3 Les primitives de securite.

Techniques utilisees : Roll-back. et Back-up.

Roll-back.

Le but est de garder une base de donnees coherentes.

Nous avons developpe deux primitives que nous avons nomme "BEGIN" et "END" qui sont destinees a encadrer toutes parties de programmes a proteger des incoherences eventuelles de la B.D. provoques par un arret intempestif de l'ordinateur (principalement des coupures de courant).

EXEMPLE :

```
<a cet instant la base de donnees est supposee coherente>
BEGIN < debut de transaction >
  Ajout d'un patient a la liste des patients (oper. 1)
  ....
  Ajout d'une tumeur a la liste des tumeurs (Oper. 2)
  <--- Coupure de courant
END < fin de transaction >
<a cet instant la base de donnees est supposee coherente>
```

Supposons que une transaction effectue deux operations. (operation 1 et 2). Si une coupure de courant survient pendant leurs executions , elle ne perturbera pas la coherence de la base de donnees. Des le retablissement de l'alimentation toutes les actions sur la B.D. effectuees pendant la transaction seront effacees.

Nom : BEGIN

Parametre : Chaine de caracteres quelconque.

Effet : Declare un debut de transaction. Toutes les modifications du contenu de la base de donnees provoqueront le sauvetage des renseignements necessaires a la

restauration eventuelle de l'etat de la B.D. dans l'etat precedent l'appel de la primitive. La chaine de caracteres est un message destine a l'utilisateur du systeme et il sera affiche a l'ecran des que l'alimentation sera retablie. Il lui indiquera par exemple qu'il doit recommencer certaines operationr.

Restriction : Un appel a une primitive BEGHN doit etre suivi par un appel a une primitive END

Nom : END

Effet : declare la fin d'une transaction. La base de donnees est supposee coherente.

Restriction : Un appel a une primitive END doit suivre un appel a une primitive BEGIN.

Back-up

Sur tous les ordinateurs les procédures de securites reposent toujours sur des hypotheses tels que la fiabilite du materiel, le comportement du systeme d'exploitation, ... La fiabilite d'un micro-ordinateur dans le contexte d'un pays en voie de developpement etant indeterminee , il est donc necessaire de prevoir des copies periodiques de la totalite ou seulement d'une partie de la base de donnees.

Nous n'avons pas construit une primitive de "Back-up" vu que tous les logiciels d'exploitation d'un systeme (CP/M par ex.) fournissent des utilitaires de copie et sauvetage de fichiers.

4.1.4. DESCRIPTION FORMELLE.

4.1.4.0 Introduction.

La description formelle suivante a pour but de permettre a un programmeur d'application d'utiliser les differentes primitives implementees en BASIC. Elle defini pour chaque primitive , le mode d'appel , les entrees , les sorties et la fonction reliant les sorties aux entrees.

Dans la meme optique nous commencerons par expliquer la structure generale d'un programme BASIC.

Nous terminerons par un exemple de l'emploi des primitives.

4.1.4.1 Structure d'un programme BASIC

Un programme basic utilisant la gestion de la base de donnees DOIT respecter une certaine structure. Deux textes l'un contenant la declaration de tableaux et l'autre le programme de gestion de la B.D. , occupent respectivement les lignes 10 a 100 et 20000 a 25000. Chaque texte est automatiquement insere dans le programme source du programmeur par la pseudo instruction "%INCLUDE filename" ou 'filename' represente la racine du nom d'un fichier. (voir schema ci-dessous).

STRUCTURE GENERALE D'UN PROGRAMME BASIC UTILISANT LA
 ----- GESTION DE LA BASE DE DONNEES -----

Numero de ligne	Contenu	Explication.
1 a 10		Non reservees.
10 11 a 100	%INCLUDE RCOM	Insertion des declarations Reservees a la declaration de certains tableaux et a l'initialisation de varia- bles propres a la gestion de la base de donnees.
101 a 19998		Non reservees.
19999	%INCLUDE ACCESBD	Insertion du texte du pro- gramme de gestion de la base de donnees.
20000 a 25000		Reservees aux routines de gestion de la B.D.
25001 a 65000		Non reservees.

 Le fichier RCOM.BAS doit contenir les instructions
 suivantes

```

10      DIM BDT(150),BDE$(150),BDKEY(15),BDKEY$(15)
15      DIM BDPTR2(5),BDKEYRW$(5),BDNREC(15)
20      DIM BDLAST(15),BDL1(50),BDRECPHYS(15)
25      DIM TBUFFER$(15),BDTIO$(15),BDZONE$(15,20)
30      DIM BDOLDZONE$(50),BDINDEX$(15,5)
35      COMMON BDT(),BDLAST(),BDRECPHYS()
40      COMMON TBUFFER$(),BDTIO$(),BDE$(),BDKEY()
45      COMMON BDKEY$(),BDNREC(),BDZONE$()
50      COMMON BDOLDZONE$(),BDINDEX(),BDREL$,BDDOM$
55      COMMON BDREL,BDDOM
60      BDBCK=20:BDTMAX=150
  
```

 Le fichier ACCESBD.BAS contient le texte du programme de
 gestion de la base de donnees.

4.1.4.2 Les primitives de description de la B.D.

Ces primitives n'ont pas fait l'objet d'une implementation en fonction directement utilisable par un programmeur. Elles ont ete integre dans un programme dont le seul role est de fournir un interface agreable entre l'utilisateur et le systeme. Si vous desirez obtenir plus de renseignements veuillez consulter le mode d'emploi du programme "DESCBD" en annexe.

4.1.4.3 Les primitives d'exploitation de la base de donnees ----- Preliminaires.

Il n'etait pas question d'ecrire des sous-routines en langage machine vu la trop forte dependance de hardware qu'elles entraînent.

Nous aurions desire pouvoir appeler une fonction par son nom et lui communiquer (recevoir) des valeurs mais le Basic dont nous disposons ne permet pas de definir des fonctions du type de celles definissable en pascal.

Ex: START ("PATIENT" , 1) effectuerait les actions definies pour la primitives START sur la premiere cle de recherche definie pour la relation "PATIENT" et retournerait un indicateur d'erreur.

N'ayant pas cette possibilite nous avons simule l'emploi de fonction :

. Le nom de la primitives a executer est communique ,au programme de gestion de la base de donnees , via une variable alpha-numerique (ACTION#) , l'indicateur d'erreur retourne par ce sous-programme est une variable entiere (FL). Les parametres sont passes via certaines variables tels que IO# , ID , CAT# , REL# , DOM# , VALEUR#.

. Le langage basic ne connait pas la notion de variables locales. De facon a eviter des interactions trop dangereuses entre les variables de l'utilisateur et celles des sous-programmes , nous avons decide d'interdire

l'utilisation de variables dont le nom commence par "BD".

Remarques generales concernant l'emploi des primitives.

Dans tous les cas , lorsque une primitive renvoie un code d'erreur different de zero , elle indique un fonctionnement incorrect et aucune des variables initialisees (REL#, DOM#, IO#, ID, NREC, et les valeurs des domaines d'une ligne lue ou initialisee par la primitive MAJDOM) lors de l'appel de la primitive ne subit de changements.

NOM : LOADBD

MODE D'APPEL : CAT#="nom de catalogue"

ACTION#="LOADBD":GOSUB 20000

ENTREES : CAT# . Chaîne non vide, de maxi. 8
caracteres. Le premier doit
etre obligatoirement alpha-
betique, les autres alpha-
numerique.
. Designe le nom d'une B.D.

SORTIES : FL Indicateur d'erreur.

FONCTION : FL=0 Si le chargement de la description de la
base de donnees de nom CAT# se termine
correctement.

FL=1 Si il n'existe pas de base de donnees
portant le nom CAT#.

FL=2 Si . le fichier destine a memoriser la
description de la base de donnees
existe, mais son contenu ne
correspond pas a ce qui est attendu
ou . La longueur du nom de la B.D. est
superieure a 8 caracteres.

REMARQUE : Le chargement de la base de données doit
preceder toutes actions. L'utilisateur veillera donc a
commencer son programme par l'appel a la primitive LOADBD
sous peine d'obtenir un code d'erreur egal a 1 lors de
l'appel des autres primitives.

NOM : OPEN

MODE D'APPEL : REL#="nom de relation":ACTION#="OPEN"

IO#="INP" ou "OUT" ou "I/O" ou "EXT"

GOSUB 20000

ENTREES : REL#

- . Nom de relation connu de la description de la base de donnees chargee lors de l'appel a la primitive "LOADBD".
- . Chaine non vide, de maxi. 7 caracteres. Le premier doit etre obligatoirement alphanumerique, les autres alphanumerique.

IO#

- . Type d'ouverture du fichier destine a enregistrer les lignes de la relation REL#.

SORTIES : FL

- . Indicateur d'erreur.

FONCTION : FL=0 Si l'accès aux lignes de la relation est permis. Les fichiers destinés à enregistrer l'extension de la relation REL# sont ouverts.

Si le mode d'ouverture choisi = "INP" alors seulement la lecture, la re-écriture, et l'effacement de lignes de l'extension de la relation REL# seront possibles.

Si le mode d'ouverture choisi = "OUT" alors seulement l'ajout de nouvelles lignes à l'extension de la relation REL# seront possibles.

Si le mode d'ouverture choisi = "EXT" alors . cette primitive efface toutes les lignes de l'extension de la relation REL#. . seulement l'ajout de nouvelles lignes à l'extension de la relation REL# seront possibles.

Si le mode d'ouverture choisi = "I/O" & le type de la relation REL# = "REL" alors l'ajout, l'effacement, la re-écriture et la lecture de

lignes de l'extension de la relation REL# seront possibles.

- FL=1 Si le nom de la relation REL# est inconnu de la description de B.D.
- FL=2 Si le type d'ouverture demande (IO#) est incompatible avec le type de la relation. (Voir tableau a la fin de la description formelle).
- FL=3 si une operation d'ouverture a deja ete effectuee pour la relation REL#.
- FL=4 si tous les buffers geres par les programmes de gestion de la B.D. sont alloues.

NOM : CLOSE

MODE D'APPEL : ACTION#="CLOSE":REL#="nom de relation"

GOSUB 20000

ENTREES : REL#

- . Nom de relation connu de la description de la base de donnees chargee lors de l'appel a la primitive "LOADBD".
- . Chaine non vide, de maxi. 7 caracteres. Le premier doit etre obligatoirement alpha-betique, les autres alpha-numerique.

SORTIES : FL

Indicateur d'erreur.

FONCTION : FL=0 Si l'operation de fermeture des fichiers destines a enregistrer l'extension de la relation se termine correctement.

FL=1 Si le nom de la relation REL# est inconnu de la description de B.D.

NOUS SOMMES OBLIGE DE DISTINGUER LE MODE D'EMPLOI DES DIFFERENTES PRIMITIVES EN FONCTION DU TYPE DE LA RELATION.

Les valeurs des codes d'erreur retournes par les primitives suivantes possedent toujours la meme signification.

- FL=1 Si le nom de la relation est inconnu de la description de la base de donnees.
- FL=2 Si le triplet (ACTION demandee, type d'ouverture declare lors de l'appel a la primitive OPEN, type de la relation) est impossible. (voir tableau a la fin de la description formelle).

De meme nous retrouverons toujours une meme variable en entree.

- REL\$. Nom de relation connu de la description de la base de donnees chargee lors de l'appel a la primitive "LOADBD".
- . Chaine non vide, de maxi. 7 caracteres. Le premier doit etre obligatoirement alphanumerique, les autres alphanumerique.

NOM : MAJDOM

MODE D'APPEL : ACTION#="MAJDOM";VALEUR#="...."

REL#="Nom de relation";DOM#="nom de domaine"

GOSUB 20000

ENTREES : REL#

DOM#

. Nom d'un domaine de la
relation REL#.

. Chaîne de 1 a 30 caracteres
alphanumeriques

VALEUR#

Chaîne de caracteres de longueur
inferieure ou egale celle decla-
ree lors de la description du
domaine DOM# de la relation REL#
(voir 4.1.4.1)

Le contenu d'un n-uplet de l'extension de la
relation REL#.

SORTIES : FL

Indicateur d'erreur.

Domaines d'un n-uplet accessibles par "READOM"

FONCTION : FL=0 Si REL# et DOM# sont connus de la descrip-

tion de la base de donnees et si

l'operation d'initialisation du

domaine DOM# d'un n-uplet de la rela-

tion REL# a la valeur de la variable

VALEUR# se termine correctement.

FL=3 si REL# ou DOM# = "" ou longueur de

VALEUR# superieure a la longueur

maximale declaree pour ce domaine.

EXEMPLE : Le court programme qui suit illustre la façon d'employer cette primitive. Nous montrerons comment initialiser les domaines "NOM" et "ADRESSE" d'un futur n-uplet de la relation PATIENT(NOM,ADRESSE) respectivement a la valeur "Martin" et "r.du cimetiere, 15"

```
ACTION$="MAJDOM":REL$="PATIENT"  
DOM$="NOM":VALEUR$="Martin":GOSUB 20000  
DOM$="ADRESSE":VALEUR$="r.du cimetiere, 15":GOSUB 20000
```

NOM : READOM

MODE D'APPEL : ACTION#="READOM"

REL#="Nom de relation":DOM#="nom de domaine"

GOSUB 20000

ENTREES : REL#

DOM# . Nom d'un domaine de la
relation REL#.
. Chaîne de 1 a 30 caracteres
alphanumeriques

Le contenu d'un n-uplet de l'extension de la
relation REL#.

SORTIES : FL

Indicateur d'erreur.

VALEUR#

Chaîne de caracteres de longueur
inferieure ou egale celle decla-
ree lors de la description du
domaine DOM# de la relation REL#
(voir 4.1.4.1)

FONCTION : FL=0 Si REL# et DOM# sont connus de la descrip-
tion de la base de donnees et si
l'operation d'initialisation de la
variable VALEUR# a la valeur du domaine
DOM# d'un n-uplet de la relation REL#
se termine correctement.

FL=3 si REL# ou DOM# = "" ou longueur de
VALEUR# superieure a la longueur
maximale declaree pour ce domaine.

EXEMPLE : Le court programme qui suit illustre la façon d'employer cette primitive. Nous montrerons comment lire la valeur des domaines "NOM" et "ADRESSE" d'un n-uplet de la relation PATIENT(NOM,ADRESSE) initialise par l'exemple precedent.

```
ACTION$="READOM":REL$="PATIENT"  
DOM$="NOM":GOSUB 20000:NOM$=VALEUR$  
DOM$="ADRESSE":GOSUB 20000:ADR$=VALEUR$
```

La valeur de la variable ...

NOM\$ --> "Martin"

ADR\$ --> "r.du cimetiere, 15"

 SI LE TYPE DE LA RELATION EST INDEXE-RELATIF OU RELATIF.

NOM : READ

MODE D'APPEL : REL\$="nom de relation":ACTION\$="READ"

NREC=...:GOSUB 20000

ENTREES : REL\$

NREC

Numero identifiant la ligne (de l'extension de la relation REL\$) a lire. Ce numero a ete attribue lors de sa creation par la primitive "WRITE".

. Valeur entiere positive, non nulle, et inferieure a 32766

SORTIES : FL

Indicateur d'erreur.

Domaines d'un n-uplet accessibles par "READOM

FONCTION : FL=0 Si l'operation d'accès a la lignes (de l'extension de la relation REL\$) identifiée par le numero NREC se termine correctement. L'utilisateur peut a present extraire les valeurs des differents domaines de la ligne par la primitive "READOM".

FL=3 Si le numero NREC n'identifie aucune ligne de l'extension de la relation.

FL=5 Si NREC < 1 ou NREC > 32766.

EXEMPLE : Soit la relation PATIENT(NOM,ADRESSE) et
l'extension :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
<2>	Niesen R.	r. Vise 652 Wand.

Le nombre entre "<>" indique un numero identifiant la ligne , attribue lors de son ajout a l'extension de la relation PATIENT par la primitive "WRITE".

L'accès a la ligne identifie par le numero 1 est realisee par la suite des instructions suivantes :

```
ACTION$="READ":REL$="PATIENT":NREC=1:GOSUB 20000
```

Dans ce cas l'indicateur d'erreur (FL) prend la valeur 0, par contre l'accès a la ligne numero 4 fournira un code de retour = 3.

Pour initialiser une variable basic (soit A\$) avec la valeur d'un domaine du n-uplet accede il vous suffit d'employer la primitive "READOM".

```
ACTION$="READOM":DOM$="NOM":GOSUB 20000:A$=VALEUR$
PRINT "--->";A$;"<---
```

L'impression du contenu de la variable A\$ donnera le resultat suivant : --->Nicolin E. <---

 SI LE TYPE DE LA RELATION EST INDEXE-RELATIF OU RELATIF.

NOM : WRITE

MODE D'APPEL : REL#="nom de relation":ACTION#="WRITE"

NREC=...:GOSUB 20000

ENTREES : REL#

NREC . Numero identifiant a attribuer
 a la ligne a ajouter a l'enten-
 sion de la relation REL#.
 . Valeur entiere positive, non
 nulle, et inferieure a 32766

Domaines d'un n-uplet initialises par "MAJDOM"

SORTIES : FL Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'ajout d'une ligne
 (de l'extension de la relation REL#),
 identifiee par le numero NREC se
 termine correctement.

FL=3 Si le numero NREC est deja attribue a une
 ligne de l'extension de la relation.

FL=5 Si NREC < 1 ou NREC > 32766.

EXEMPLE : Soit la relation PATIENT(NOM,ADRESSE) et
l'extension :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
<2>	Niesen R.	r. Vise 652 Wand.

Le nombre entre "<>" indique un numero identifiant la ligne , attribue lors de son ajout a l'extension de la relation PATIENT par la primitive "WRITE".

L'ajout d'une ligne identifiee par le numero 3 se compose de deux etapes.

(1) la preparation du contenu de la future ligne.

Soit la future ligne :

("Servais B. ", "r.j.Wauters 110 Viv. ")

(2) Attribution d'un numero identifiant et creation.

Ces deux etapes se traduisent par la suite des instructions suivantes :

```
; Premiere etape
ACTION#="MAJDOM":REL#="PATIENT"
DOM#="NOM":VALEUR#="Servais B.":GOSUB 20000
DOM#="ADRESSE":VALEUR#="r.j.Wauters 110 viv.":GOSUB 20000
; Seconde etape
ACTION#="WRITE":NREC=3:GOSUB 20000
```

L'extension de la relation PATIENT est maintenant la suivante :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
<2>	Niesen R.	r. Vise 652 Wand.
---> <3>	Servais B.	r.j.Wauters 110 Viv.

SI LE TYPE DE LA RELATION EST RELATIF.

NOM : REWRITE

MODE D'APPEL : REL#="nom de relation":ACTION#="REWRITE"

NREC=...:GOSUB 20000

ENTREES : REL#

NREC . Numero identifiant la ligne
(de l'extension de la relation)
a modifier.

. Valeur entiere positive, non
nulle, et inferieure a 32766

Domaines d'un n-uplet initialises par "MAJDOM"

SORTIES : FL . Indicateur d'erreur.

FONCTION : FL=0 Si l'operation de reecriture d'une ligne
(de l'extension de la relation REL#),
identifiee par le numero NREC se
termine correctement.

FL=3 Si le numero NREC n'identifie aucune
ligne de l'extension de la relation.

FL=5 Si NREC < 1 ou NREC > 32766.

EXEMPLE : Soit la relation PATIENT(NOM,ADRESSE) et
l'extension :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
<2>	Niesen R.	r. Vise 652 Wand.

Le nombre entre "<>" indique un numero identifiant la ligne , attribue lors de son ajout a l'extension de la relation PATIENT par la primitive "WRITE".

La reecriture d'une ligne identifie par le numero 2 se compose de deux etapes.

(1) la preparation du nouveau contenu de la ligne.

("Servais B. ", "r.j.Wauters 110 Viv. ")

(2) Reecriture

Ces deux etapes se traduisent par la suite des instructions suivantes :

```
; Premiere etape
ACTION$="MAJDOM":REL$="PATIENT"
DOM$="NOM":VALEUR$="Servais B.":GOSUB 20000
DOM$="ADRESSE":VALEUR$="r.j.Wauters 110 viv.":GOSUB 20000
; Seconde etape
ACTION$="REWRITE":NREC=2:GOSUB 20000
```

L'extension de la relation PATIENT est maintenant la suivante :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
---> <2>	Servais B.	r.j.Wauters 110 Viv.

SI LE TYPE DE LA RELATION EST RELATIF.

NOM : DELETE

MODE D'APPEL : REL#="nom de relation":ACTION#="DELETE"

NREC=...:GOSUB 20000

ENTREES : REL#

NREC

. Numero identifiant la ligne
(de l'extension de la relation)
a effacer.

. Valeur entiere positive, non
nulle, et inferieure a 32766

SORTIES : FL

Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'effacement d'une ligne
(de l'extension de la relation REL#),
identifiee par le numero NREC se
termine correctement.

FL=3 Si le numero NREC n'identifie aucune
ligne de l'extension de la relation.

FL=5 Si NREC < 1 ou NREC > 32766.

EXEMPLE : Soit la relation PATIENT(NOM,ADRESSE) et
l'extension :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.
<2>	Niesen R.	r. Vise 652 Wand.

Le nombre entre "<>" indique un numero identifiant la ligne , attribue lors de son ajout a l'extension de la relation PATIENT par la primitive "WRITE".

L'effacement d'une ligne identifie par le numero 2 se traduit par les instructions suivantes :

```
REL$="PATIENT":ACTION$="DELETE":NREC=2:GOSUB 20000
```

L'extension de la relation PATIENT est maintenant la suivante :

	NOM	ADRESSE
<1>	Nicolin E.	ch. de Tongres 616 Roc.

 SI LE TYPE DE LA RELATION EST
 INDEXE-RELATIF ou INDEXE-SEQUENTIEL

NOM : START

MODE D'APPEL : REL\$="nom de relation":ACTION\$="START"

ID=...:GOSUB 20000

ENTREES : REL\$

ID . Numero de cle de recherche
 . Valeur entiere positive, non
 nulle, et inferieure a 6.

Domaines d'un n-uplet initialises par "MAJDOM"

SORTIES : FL Indicateur d'erreur.

Domaines d'un n-uplet accessibles pas "READOM"

FONCTION : FL=0 Si l'operation de recherche d'une ligne
 (de l'extension de la relation REL\$),
 se termine correctement. La ligne
 trouvee possede une cle directement
 superieure a la valeur de la recherche
 constituee par la concatenation des
 valeurs des domaines (du n-uplet en
 entree) possedant un numero d'apparte-
 nance egal a ID.

Exception : si cette cle de recherche
 initiale est superieure a toutes cles
 des lignes de l'extension de la rela-
 tion alors la ligne trouvee par le
 "START" sera celle de la relation
 dont la valeur de cle est la plus

grande.

Les valeurs des domaines du n-uplet
trouve sont accessible via
la primitive "READOM".

FL=6 Si ID < 1 ou ID > 5

FL=7 Si le numero de cle de recherche (ID)
n'est pas defini pour la relation REL#.

FL=8 Si l'extension de la relation = vide.

EXEMPLE : Soit la relation PATIENT(NOM,ADRESSE) , et
l'extension :

NOM	ADRESSE
Nizet Th.	r.Acacias 31 Hermee.
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.
Firon M.	r. Pommiers 9

Nous supposerons que la premiere cle de recherche se
compose du NOM (c-a-d que le numero d'appartenance attribue
au domaine NOM de la relation PATIENT pendant sa definition
(voir 4.1.4.2) est egal a un).

La recherche d'une ligne au moyen de certains criteres se
decompose en deux etapes.

(1) la preparation de la cle de recherche. C-a-d
initialiser les valeurs des domaines de la cle de recherche
a une certaine chaine de caracteres. Par exemple ,
initialiser le domaine "NOM" d'un n-uplet de la relation
PATIENT a la chaine de caracteres : "Nilet"

(2) Recherche

L'extension de la relation PATIENT ne change pas et le programme de gestion de la base de données vous fournira la ligne dont la cle est supérieure ou égale à la cle de recherche. c-a-d

```
("Nizet Th.      ", "r.Acacias 31 Hermee.")
```

Exception , lorsque la cle de recherche est supérieure à la cle de toutes lignes alors la primitive fournit la ligne de la relation dont la valeur de la cle est la plus grande.

Ex: Si la valeur du domaine NOM du n-uplet est "xxxxxxxxxxxxx" alors la ligne trouvée sera :

```
("Piron M.      ", "r. Pommiers 9      ")
```

Les deux étapes précitées se traduisent par la suite des instructions suivantes :

```
; Première étape
ACTION$="MAJDOM":REL$="PATIENT"
DOM$="NOM":VALEUR$="Nilet":GOSUB 20000
; Seconde étape
ACTION$="START":ID=1:GOSUB 20000
```

La valeur des domaines du n-uplet trouvés sont accessibles via la primitive "READOM" de la façon suivante :

```
ACTION$="READOM":REL$="PATIENT"
DOM$="NOM":GOSUB 20000:NOM$=VALEUR$
DOM$="ADRESSE":GOSUB 20000:ADRESSE$=VALEUR$
```

Le contenu des variables NOM\$ et ADRESSE\$ sont respectivement "Nizet Th. " et "r.Acacias 31 Hermee."

SI LE TYPE DE LA RELATION EST
INDEXE-RELATIF ou INDEXE-SEQUENTIEL

NOM : READNEXT

MODE D'APPEL : REL#="nom de relation":ACTION#="READNEXT"

ENTREES : REL#

SORTIE : FL Indicateur d'erreur.

Domaines d'un n-uplet accessibles par "READOM"

FONCTION : FL=0 Si l'operation d'accès a la ligne suivante
se termine correctement. Les valeurs
des domaines du n-uplet trouve sont
accessible via la primitive "READOM".

FL=3 Si il n'existe plus de lignes de
l'extension de la relation qui soit
superieur a la derniere cle du n-uplet
trouve par une operation START ou
READNEXT precedente.

FL=9 Si l'operation READNEXT n'est pas precedee
par un START ou un READNEXT.

EXEMPLE : Considerons l'exemple precedent (START). Nous avons trouve la ligne :

```
("Nizet Th.      ",      "r.Acacias 31  Hermee.")
```

Le court programme qui suit illustre la facon d'accéder aux différentes lignes de la relation PATIENT dans l'ordre croissant des NOMs a partir du nom de patient "Nizet Th".

```
PRINT "NOM          ADRESSE"
ACTION$="MAJDOM":REL$="PATIENT"
DOM$="NOM":VALEUR$="Nizet":GOSUB 20000
ACTION$="START":ID=1:GOSUB 20000
WHILE FL<>3 AND FL<>8
  ACTION$="READOM"
  DOM$="NOM":GOSUB 20000:PRINT VALEUR$;
  DOM$="ADRESSE":GOSUB 20000:PRINT VALEUR$
  ACTION$="READNEXT":GOSUB 20000
WEND
```

Il imprime le texte suivant :

```
NOM          ADRESSE
Nizet Th.    r.Acacias 31 Hermee.
Piron M.     r. Pommiers 9
```

SI LE TYPE DE LA RELATION EST
INDEXE-RELATIF ou INDEXE-SEQUENTIEL

NOM : DELETE

MODE D'APPEL : REL#="nom de relation":ACTION#="DELETE"

ENTREES : REL#

SORTIE : FL

Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'effacement de la ligne
(de l'extension de la relation REL#)
dernierement accedee par une primitive
"START" ou "READNEXT" se termine
correctement.

FL=4 Si l'operation d'effacement n'est pas
precedee par un START ou READNEXT.

EXEMPLE : Considerons l'exemple precedent (START). Nous avons trouve la ligne :

("Nizet Th. ", "r.Acacias 31 Hermee.")

Le court programme qui suit illustre la facon d'accéder aux différentes lignes de la relation PATIENT dans l'ordre croissant des NOMs a partir du nom de patient "Nizet Th" et la methode d'effacement des lignes trouvees.

```

ACTION$="MAJDOM":REL$="PATIENT"
DOM$="NOM":VALEUR$="Nizet":GOSUB 20000
ACTION$="START":ID=1:GOSUB 20000
WHILE FL<>3 AND FL<>8
    ACTION$="DELETE":GOSUB 20000
    ACTION$="READNEXT":GOSUB 20000
WEND

```

Extension de la relation "PATIENT" AVANT l'effacement de certaines lignes :

	NOM	ADRESSE
---	Nizet Th.	r.Acacias 31 Hermee.
	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.
---	Piron M.	r. Pommiers 9

Extension de la relation "PATIENT" APRES l'effacement de certaines lignes :

	NOM	ADRESSE
	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.

SI LE TYPE DE LA RELATION EST
INDEXE-RELATIF ou INDEXE-SEQUENTIEL

NOM : REWRITE

MODE D'APPEL : REL\$="nom de relation":ACTION\$="REWRITE"

ENTREES : REL\$

Le contenu d'un n-uplet de l'extension de la
relation REL\$. Ce n-uplet sera initialise soit
par l'utilisation de la primitive "MAJDOM" ou
par une des primitives START, READNEXT ou READ.

SORTIE : FL Indicateur d'erreur.

FONCTION : FL=0 Si l'operation de reecriture de la ligne
de l'extension de la relation REL\$
(dernierement accedee par une primitive
"START" ou "READNEXT") se termine
correctement.

FL=4 Si l'operation de reecriture n'est pas
precedee par un START ou READNEXT.

EXEMPLE : Considerons l'exemple precedent (START). Nous avons trouve la ligne :

("Nizet Th. ", "r.Acacias 31 Hermee.")

Nous allons remplacer le contenu de la ligne ainsi trouvee par la ligne suivante :

("Deruelle M. ", "r.Acacias 31 Heree. ").

Ce remplacement s'effectue en trois etapes :

- (1) Acces a une ligne a remplacer
- (2) Initialisation d'une nouvelle ligne
- (3) Remplacement de l'ancienne ligne par une nouvelle

```
; Acces a la ligne a remplacer
ACTION#="MAJDOM":REL#="PATIENT"
DOM#="NOM":VALEUR#="Nizet":GOSUB 20000
ACTION#="START":ID=1:GOSUB 20000
; Initialisation d'une nouvelle ligne
ACTION#="MAJDOM":VALEUR#="Deruelle M.":GOSUB 20000
; Reecriture
ACTION#="REWRITE":GOSUB 20000
```

Extension de la relation "PATIENT" AVANT le remplacement de la ligne precitee :

	NOM	ADRESSE
--->	Nizet Th.	r.Acacias 31 Hermee.
	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.
	Piron M.	r. Pommiers 9

Extension de la relation "PATIENT" APRES le remplacement :

	NOM	ADRESSE
---->	Deruelle M.	r.Acacias 31 Hermee.
	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.
	Piron M.	r. Pommiers 9

SI LE TYPE DE LA RELATION EST INDEXE-SEQUENTIEL

NOM : WRITE

MODE D'APPEL : REL\$="nom de relation":ACTION\$="WRITE"

ENTREES : REL\$

Le contenu d'un n-uplet de l'extension de la relation REL\$. Ce n-uplet sera initialise soit par l'utilisation de la primitive "MAJDOM" ou par une des primitives START, READNEXT.

SORTIE : FL Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'ajout d'une ligne a l'extension de la relation REL\$ se termine normalement.

FL=5 Si l'ajout est impossible par manque de place dans les fichiers destines a enregistrer les lignes de la relation REL\$. c-a-d qu'il y a eu plus de 32765 lignes crees.

EXEMPLE : Soit l'extension de la relation "PATIENT" avant l'ajout d'une ligne :

NOM	ADRESSE
Nizet Th.	r.Acacias 31 Hermee.
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.
Piron M.	r. Pommiers 9

L'ajout de la ligne suivante :

("Nizet Th. ", "r.S.PANIS 30 Crisnee ").

s'effectue en deux etapes :

- (1) Initialisation d'une nouvelle ligne
- (2) L'appel a la primitive d'ajout.

Le programme correspondant :

```
; premiere etape
ACTION#="MAJDOM":REL#="PATIENT"
DOM#="NOM":VALEUR#="Nizet Th.":GOSUB 20000
DOM#="ADRESSE":VALEUR#="r.S.PANIS 30 Crisnee":GOSUB 20000
; seconde etape
ACTION#="WRITE":GOSUB 20000
```

L'extension de la relation est maintenant la suivante :

NOM	ADRESSE
Nizet Th.	r.Acacias 31 Hermee.
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.
Piron M.	r. Pommiers 9
---> Nizet Th.	r.S.PANIS 30 Crisnee

L'accès via la première cle de recherche (voir Exemple du START et du READNEXT) aux différentes lignes de l'extension de la relation "PATIENT" donnera le résultat suivant :

NOM	ADRESSE
Lange N.	r. Vise 652 Wand.
Moreau E.	ch. de Tongres 616 Roc.
---> Nizet Th.	r.S.PANIS 30 Crisnee
---> Nizet Th.	r.Acacias 31 Hermee.
Piron M.	r. Pommiers 9

Vous remarquerez que l'accès aux lignes possédant la même cle s'effectue selon l'ordre LIFO (Last In First Out).

SI LE TYPE DE LA RELATION EST SEQUENTIEL

NOM : READ

MODE D'APPEL : REL#="nom de relation":ACTION#="READ"

ENTREES : REL#

SORTIE : FL Indicateur d'erreur.

Domaines d'un n-uplet accessibles par "READOM"

FONCTION : FL=0 Si l'operation d'accès sequentiel (selon l'ordre de creation par la primitive "WRITE") a une ligne de l'extension de la relation REL# se termine normalement. Les valeurs des domaines du n-uplet accede sont accessibles via la primitive "READOM".

FL=3 Si plus (ou pas) de lignes a lire.

EXEMPLE : Supposons que les lignes de l'extension de la relation "PATIENT" ont été créées selon l'ordre suivant :

("Nizet Th. ", "r.Acacias 31 Hermee. ")

("Moreau E. ", "ch. de Tongres 616 Roc.")

("Lange N. ", "r. Vise 652 Wand. ")

La lecture séquentielle s'effectue à partir de la première ligne de la relation jusqu'à la dernière.

Dans l'exemple suivant, nous accédons à chaque ligne de la relation "PATIENT" et nous imprimons la valeur des domaines du n-uplet.

```
; Acces a la premiere ligne
REL$="PATIENT":ACTION$="READ":GOSUB 20000
WHILE FL<>3
    ACTION$="READOM"
    DOM$="NOM":GOSUB 20000:PRINT "NOM ";VALEUR$
    DOM$="ADRESSE":GOSUB 20000:PRINT "ADRESSE ";VALEUR$
    ; Acces a la ligne suivante
    ACTION$="READ":GOSUB 20000
WEND
```

SI LE TYPE DE LA RELATION EST SEQUENTIEL

NOM : WRITE

MODE D'APPEL : REL#="nom de relation":ACTION#="WRITE"

ENTREES : REL#

Le contenu d'un n-uplet de l'extension de la relation REL#. Ce n-uplet sera initialise soit par l'utilisation de la primitive "MAJDOM" ou par la primitive READ.

SORTIE : FL

Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'ajout d'une ligne a l'extension de la relation REL# se termine normalement.

FL=5 Si l'ajout est impossible par manque de place dans les fichiers destines a enregistrer les lignes de la relation REL#. Il y a eu plus de 32765 lignes crees.

EXEMPLE : Soit l'extension de la relation "PATIENT" avant l'ajout d'une ligne :

NOM	ADRESSE
Nizet Th.	r.Acacias 31 Hermee.
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.

L'ajout de la ligne suivante :

("Nizet Th. ", "r.S.PANIS 30 Crisnee ").

s'effectue en deux etapes :

- (1) Initialisation d'une nouvelle ligne
- (2) L'appel a la primitive d'ajout.

Le programme correspondant :

```

; premiere etape
ACTION$="MAJDOM":REL$="PATIENT"
DOM$="NOM":VALEUR$="Nizet Th.":GOSUB 20000
DOM$="ADRESSE":VALEUR$="r.S.PANIS 30 Crisnee":GOSUB 20000
; seconde etape
ACTION$="WRITE":GOSUB 20000

```

L'extension de la relation est maintenant la suivante :

	NOM	ADRESSE
	Nizet Th.	r.Acacias 31 Hermee.
	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.
--->	Nizet Th.	r.S.PANIS 30 Crisnee

SI LE TYPE DE LA RELATION EST SEQUENTIEL

NOM : DELETE

MODE D'APPEL : REL#="nom de relation":ACTION#="DELETE"

ENTREES : REL#

SORTIE : FL Indicateur d'erreur.

FONCTION : FL=0 Si l'operation d'effacement de la ligne
(de l'extension de la relation REL#)
dernierement accedee par une primitive
"READ" se termine correctement.

FL=4 Si l'operation d'effacement n'est pas
precedee par "READ"

EXEMPLE : Soit l'extension de la relation "PATIENT" avant l'effacement d'une ligne :

NOM	ADRESSE
Nizet Th.	r.Acacias 31 Hermee.
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.
Nizet Th.	r.S.PANIS 30 Crisnee

Le programme suivant efface le n-uplet dont la valeur du domaine NOM commence par la chaine de caracteres "Nizet".

```

; Acces a la premiere ligne
CLE$="Nizet":L=LEN(CLE$):' longueur de la variable CLE$
REL$="PATIENT":ACTION$="READ":GOSUB 20000
WHILE FL<>3
    ACTION$="READOM"
    DOM$="NOM":GOSUB 20000:NOM$=VALEUR$
    IF LEFT$(NOM$,L)=CLE$
        THEN ACTION$="DELETE":GOSUB 20000:' effacement
    ; Acces a la ligne suivante
    ACTION$="READ":GOSUB 20000
WEND

```

L'extension de la relation est maintenant la suivante :

NOM	ADRESSE
Moreau E.	ch. de Tongres 616 Roc.
Lange N.	r. Vise 652 Wand.

 SI LE TYPE DE LA RELATION EST SEQUENTIEL

NOM : REWRITE

MODE D'APPEL : REL#="nom de relation":ACTION#="REWRITE"

ENTREES : REL#

Le contenu d'un n-uplet de l'extension de la relation REL#. Ce n-uplet sera initialise soit par l'utilisation de la primitive "MAJDOM" ou par la primitive READ.

SORTIE : FL Indicateur d'erreur.

FONCTION : FL=0 Si l'operation de reecriture de la ligne (de l'extension de la relation REL#) derniere accedee par une primitive "READ" se termine correctement.

FL=4 Si l'operation de reecriture n'est pas precedee par "READ"

EXEMPLE : Soit l'extension de la relation "PATIENT" :

NOM	ADRESSE
Nizet Th.	r. Acacias 31 Hermee.
Lange N.	r. Vise 652 Wand.
Piron M.	r. Pommiers 9

Le programme ci-dessous remplace la premiere ligne de l'extension de la relation REL\$ par la ligne suivante :

("Moreau E. ", "ch. de Tongres 616 Roc. ")

Ce remplacement s'effectue en trois etapes :

- (1) Acces a une ligne a remplacer
- (2) Initialisation d'une nouvelle ligne
- (3) Remplacement de l'ancienne ligne par une nouvelle

```
; Acces a une ligne (par exemple la premiere)
REL$="PATIENT":ACTION$="READ":GOSUB 20000
; Initialisation de la nouvelle ligne
ACTION$="MAJDOM"
DOM$="NOM":VALEUR$="Moreau E.":GOSUB 20000
DOM$="ADRESSE":VALEUR$="ch.de Tongres 616 Roc."
GOSUB 20000
; Remplacement de la ligne derniere accedee
ACTION$="REWRITE":GOSUB 20000
```

Extension de la relation apres le remplacement de la premiere ligne :

	NOM	ADRESSE
--->	Moreau E.	ch. de Tongres 616 Roc.
	Lange N.	r. Vise 652 Wand.
	Piron M.	r. Pommiers 9

4.1.4.3 les primitives de securites.

NOM : BEGIN

MODE D'APPEL : ACTION#="BEGIN":MESS#="...":GOSUB 20000

ENTREES : MESS#

SORTIES : /

FONCTION : Declare un debut de transaction.

Si une transaction est deja en cours et que vous appelez cette primitive alors le message suivant sera affiche :
"Transaction deja en cours !" et le programme s'arretera.

NOM : END

MODE D'APPEL : ACTION#="END":GOSUB 20000

ENTREE : /

SORTIE : /

FONCTION : Declare une fin de transaction.

Si une transaction n'est pas en cours et que vous appelez cette primitive alors le message suivant sera affiche :
"Pas de transaction en cours !" et le programme s'arretera.

EXEMPLE : Soit une relation PATIENT(NOM,ADRESSE)

Le court programmes qui suit illustre la facon d'accéder aux différentes lignes de la relation PATIENT dans l'ordre croissant des NOMs a partir du nom de patient "Nizet Th" et expose la methode d'effacement des lignes trouvees tout en protegeant la base de donnees d'une coupure eventuelle de courant.

Si cet evenement intervient pendant l'execution du programme nous desirons que le message suivant soit affiche : "Recommencez l'effacement s-v-p"

```

MESS$="Recommencez l'effacement s-v-p":GOSUB 20000
ACTION$="OPEN":ID$="INF":REL$="PATIENT":GOSUB 20000
ACTION$="MAJDOM"
DOM$="NOM":VALEUR$="Nizet":GOSUB 20000
ACTION$="START":ID=1:GOSUB 20000
WHILE FL<>3 AND FL<>8
    ACTION$="DELETE":GOSUB 20000
    ACTION$="READNEXT":GOSUB 20000
WEND
ACTION$="END":GOSUB 20000
ACTION$="CLOSE":GOSUB 20000

```

 TABLE DES DIFFERENTES VARIABLES D'ENTREES QUI DOIVENT ETRE
 INITIALISEES AVANT L'APPEL DE CERTAINES FONCTIONS.

Nom de fonction -->	START	READNEXT	REWRITE	OPEN	CLOSE
	DELETE	WRITE	READ		
TYPE DE RELATION	+	+	+	+	+
Sequential (SEQ)	!	x	!	x	!
Relatif (REL)	!	x	!	x	!
Indexe-Relatif (IDR)	!	x	!	x	!
Indexe-sequent. (IDX)	!	x	!	x	!
VARIABLES en ENTREE	+	+	+	+	+
REL#	!	xx	!	xx	!
ID	!	x	!		!
NREC	!	x	!	x	!
IO#	!		!	x	!

TABLE DES TRIPLETS

(Type d'ouverture choisi lors de l'OPEN, Type de relation, operation) POSSIBLES.

Nom de fonction -->	START	READNEXT	REWRITE	OPEN
	DELETE	WRITE	READ	
TYPE D'OUVERTURE	+	+	+	+
INP (input)	!	x	!	x
OUT (output)	!		!	x
I/O (input-output)	!	x	!	x
EXT (extend)	!		!	x
TYPE DE RELATION	+	+	+	+
Sequential (SEQ)	!	x	!	xxx
Relatif (REL)	!	xx	!	xxx
Indexe-Relatif (IDR)	!	x	!	xx
Indexe-sequent. (IDX)	!	x	!	xx

4.1.5 Description technique

4.1.5.0 Introduction.

Nous commencerons par une description des fichiers , variables et tableaux principaux utilises pour l'implementation des primitives. Elle sera suivie par une explication d'un schema des transferts de donnees provoques par l'appel des primitives.

Nous aurions desire termine par une description detaillee des principaux algorithmes de facon a permettre a un programmeur d'eventuellement ajouter ou modifier des primitives mais comme dans les mois qui suivent nous pensons changer radicalement leurs implementations , nous avons trouve inutile de fournir trop de details. Nous montrerons seulement l'effet de certains algorithmes sur le contenu des differents fichiers et tableaux utilises.

4.1.5.1 Description de la structure des fichiers.

a) Structure des fichiers destines a memoriser les lignes des relations.

Nous avons a notre disposition un langage de programmation (le BASIC) qui ne permettait que deux types de fichiers, le sequentiel et le relatif (random). De facon a ne pas confondre avec les types de fichiers sites precedemment, nous conviendrons d'appeler respectivement sequentiel-basic et relatif-basic les fichiers directement implementes en basic sans passer par la gestion de la base de donnees.

Les fichiers sequentiels-basic imposent une ecriture et une lecture sequentielle des enregistrements et il n'est pas possible d'ajouter des informations a la fin de ce fichier sans etre obliger de le lire entierement. De plus nous avons decouvert, par hasard, qu'un fichier sequentiel ouvert en ecriture etait extremement vulnérable. Un arret imprevu d'un programme utilisant ce type de fichier provoque irremediablement la perte des informations enregistrees.

Les fichiers relatifs-basic peuvent etre consideres comme une suite de cases de longueur identique, numerotees de 1 a 32769, directement accessible aussi bien en lecture que en ecriture. Nous avons egalement decouvert qu'ils sont plus fiables que les fichiers sequentiels-basic; toutes

informations effectivement ecrites sur le support magnetique sont recuperables en cas d'incident.

Nous choisirons d'implementer tous les fichiers de la base de donnees sous la forme de fichiers relatifs-basic.

Les differents types d'enregistrement physique.

- Enregistrement de ligne.

Un enregistrement de ce type correspond a une ligne d'une relation.

La longueur des enregistrements d'un fichier (LONG) correspondant a une relation est fixe et egale a la somme des longueurs des differents domaines specifiques pour une relation plus un.

Par exemple : soit une relation PATIENT (NOM , ADRESSE)

La longueur maximale du NOM et de l'ADRESSE etant respectivement de 15 et 30 caracteres , on peut en deduire la longueur des enregistrements $15 + 30 + 1 = 46$ caracteres.

Le premier byte (FLAG) de cet enregistrement peut prendre deux valeurs 0 et 255 et indiquent respectivement l'absence et la presence de valeurs significatives dans les bytes qui suivent.

```

I--I--+-+-+-+-+.... -+-+-I
FLAG
<----- LONG ---->
zone destinee a contenir une ligne d'une
relation.
```

- Enregistrement initial

De longueur egale a la longueur (LONG) des enregistrements de ligne plus un , cet enregistrement est le premier et fournit un pointeur (LREC) vers le dernier record du fichier.

Les valeurs possibles sont 0,2,3,...,32766. La valeur nulle indique un fichier vide, la valeur 1 est exclue vu que le premier enregistrement est particulier.

```

I--+--I--+--+--+--+--+.... --+--+--I
<LREC>
  <----- LONG ----->

```

Le type sequentiel (SEQ).
 - - - - -

A une relation de type SEQuentiel (SEQ) correspond un seul fichier. Il est constitue d'un enregistrement initial suivi de 0..32766 enregistrements de ligne.

Les types Indexe. (IDX et IDR)
 - - - - -

A une relation de type Indexe sequentiel correspond au moins deux fichiers , le fichier principal et au moins un fichier secondaire (ce dernier sera aussi appele "fichier d'index").

Il y a autant de fichiers secondaires qu'il y a de cles de recherche definies pour la relation. Le fichier secondaire est un fichier inverse. Il contient une liste de pointeurs. Le premier indique le dernier record ecrit dans ce fichier , les autres permettent d'accéder au fichier principal selon l'ordre croissant des cles.

L'IDEE PRINCIPALE est d'effectuer une recherche dichotomique dans le fichier principal via le fichier secondaire. Ce fichier secondaire permet donc de voir le fichier principal comme trie sur la cle de recherche.

Le fichier principal en indexe-sequentiel est constitue d'un enregistrement initial suivi de 0 .. 32768

enregistrements de lignes ,le fichier principal en
indexe-relatif a la meme structure, exepte qu'il ne possede
pas de premier enregistrement initial.

le type relatif (RELatif)

- - - - -

A une relation du type RELatif correspond un seul fichier.
Il contient de 0 .. 32766 enregistrements de ligne.

Exemple : la structure d'un fichier du type IDX.

=====

Soit une relation :

	R	(A	,	B	,	C)
cle de recherche	<1	-----	1>		<2	-----	2>	
	"abc"		"def"		"z "			
	"bcd"		"aaa"		"ccc"			
	"eee"		"eee"		"ddd"			
	"aaa"		"ab "		"cd "			

	index 1	index 2	fichier principal du type IDX.												
	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+	+--+--+
1>	I 5 I	I 5 I	I	5	!										I
2>	I 5 I	I 3 I	I255a	b	c	!	d	e	f	!	z	z	z	z	I
3>	I 2 I	I 5 I	I255b	c	d	!	a	a	a	!	c	c	c	c	I
4>	I 3 I	I 4 I	I255e	e	e	!	e	e	e	!	d	d	d	d	I
5>	I 4 .I	I 2 I	I255a	a	a	!	a	d		!	c	d			I
6>	I 0 I	I 0 I	I 0												I
7>

Fichiers :

R.ID1	R.ID2	R.IDX
-------	-------	-------

Les indexs contiennent une liste de pointeurs vers le fichier principal.

Le fichier principal est compose d'un enregistrement initial suivi de 0..32766 enregistrements de ligne. Le premier byte de chaque enregistrement de ligne indique l'absence ou la presence d'informations dans les bytes qui suivent.

(b) Structure des fichiers destines au sauvetage et a la
restauration de la base de donnees.

Deux fichiers , de longueur fixe et de type relatif-basic
seront crees ou remplaces lors du sauvetage de la
description de la base de donnees.

Le PREMIER ne contiendra que des nombres entiers. Les trois
premiers nombres ont une signification particuliere et les
suivants refletent le contenu de la table BDT (voir
4.1.5.2). Le nom de ce fichier est compose du nom de la
base de donnees suivi par un suffixe ".NUM".

Le premier nombre indique la Longueur du plus long
descripteur dans la table BDE#, Le second la valeur de la
variable DEPT et le troisieme la valeur de la variable
BDTMAX.

Le SECOND fichier enregistre les DEPT chaines de caracteres
de la table BDE# dans des enregistrements de longueur egale
a celle du plus long descripteur plus un. Le premier byte
de chaque enregistrement indique la longueur du descripteur
qu'il contient. Le nom du fichier est compose du nom de la
base de donnees suivi par un suffixe ".ALP".

4.1.5.2 Descripteurs des variables et tableaux

Les primitives de descriptions de la B.D. se base sur une structure de donnees composee de descripteurs. Ils contiendront ,sous une forme condensee et structuree , des renseignements permettant d'accéder avec rapidites a une description d'une relation, a une description d'un domaine d'une relation , a la valeur d'un domaine, et d'ajouter aisement de nouveaux domaines a une relation.

Descripteurs.

a) Tableaux et variables particulieres utilisees pour memoriser les descripteurs.

BDE# : Table de chaines de caracteres, a une dimension.

Elle contient les descripteurs.

BDT : Table d'entiers positifs, a une dimension.

#BDT = 0 .. BDTMAX.

BDTMAX: Nombre d'entrees dans la table BDE# et BDT

BDEPT : Pointeur vers la premiere entree libre de BDE#.

0 <= BDEPT <= BDTMAX

b) Descripteur de base de donnees

```

I-----!-----!-----!-----!-----+-----+-----+-----+-----+-----+-----+
I LCAT   ! NBREL  ! ACTREL !  F R E D O M   ! NOMCAT   I
I-----!-----!-----!-----!-----+-----+-----+-----+-----+-----+-----+

```

LCAT : Longueur du nom de la base de donnees NOMCAT.

Longueur : 1 byte.

Valeurs possibles : $0 < \text{LCAT} < 9$.

NBREL : Nombre maximum de relations qui peuvent etre decrites. Longueur : 1 byte.

Valeurs possibles : $0 < \text{NBREL} < 255$

ACTREL : Nombre de relations actuellement definies dans la base de donnees. Longueur : 1 byte.

Valeurs possibles : $0 \leq \text{ACTREL} \leq \text{NBREL}$

NOMCAT : Nom donne au catalogue par l'utilisateur. C'est egalement le nom de la base de donnees.

Longueur : LCAT bytes.

Valeurs possibles : Chaine de caracteres (A..Z).

FREDOM : Pointeur dans la table BDT ,permettant d'accéder a la premiere zone non reservee de celle-ci.

Longueur : 2 bytes.

Valeurs possibles : $0 \leq \text{FREDOM} \leq \text{BDTMAX}$

c) Descripteur d'une relation.

```

I-----+-----+-----+-----+...
I LREL   ! TYPEACCES           ! DEGMAX ! ACTDEG !
I-----+-----+-----+-----+...

...!-----+-----+-----+-----+I
!      PTDOM           !   NOMREL           !
...!-----+-----+-----+-----+I

```

LREL : Longueur du nom de la relation NOMREL

Longueur : 1 byte.

Valeurs possibles : $0 < LREL < 9$.

TYPEACCES : Type d'accès aux lignes de la relation (type de fichier). Longueur : 3 bytes.

Valeurs possibles "REL", "SEQ", "IDR", "IDX"

DEGMAX : Degré maximum de la relation.

Longueur : 1 byte

Valeurs possibles : $0 < DEGMAX < BDTMAX - FREDOM$.

ACTDEG : Degré actuel de la relation.

Longueur : 1 byte

Valeurs possibles : $0 < ACTDEG \leq DEGMAX$

PTDOM : Pointeur vers la table BDT permettant d'accéder, via une indirection, au premier domaine de la relation. $BDE\#(BDT(PTDOM))$ donnera le premier descripteur de domaine de la relation, $BDE\#(BDT(PTDOM+1))$ fournira le suivant ...

Longueur : 2 bytes.

Valeurs possibles : $2 \leq \text{PTDOM} \leq \text{BDTMAX}$

NOMREL : Nom de la relation.

Longueur : LREL bytes.

Valeurs possibles : Suite de caracteres (A..Z).

d) Descripteur de domaine

```

I-----|-----|-----|-----|-----|...
I LDOM   | IDENT  | LMAXVAL | LACTVAL | VALEUR  |
I-----|-----|-----|-----|-----|I...

```

```

I----- ..... -----I
I  NOMDOM                               I
I----- ..... -----I

```

LDOM : Longueur du nom du domaine NOMDOM.

Longueur : 1 byte.

Valeurs possibles : $0 < LDOM < 30$.

IDENT : La valeur de ce byte permet de déterminer les numeros d'appartenance du domaine a une cle de recherche.

Valeurs possibles : $0 \leq IDENT < 32$

Chaque Bit indique par sa position un numero d'appartance.

Exemple : si IDENT a la valeur 5.

Nous obtiendrons la configuration suivante :

```

bit      8 7 6 5 4 3 2 1
         - - - 0 0 1 0 1

```

(les tirets indique les bits inutilises.)

Elle indique que le domaine appartient a la premiere et a la troisieme cle de recherche.

La valeur 0 indique donc qu'un domaine ne fait pas partie d'une cle de recherche.

LMAXVAL : Longueur maximale de la valeur du domaine.

Longueur : 1 byte.

Valeurs possibles : $1 \leq LMAXVAL \leq 60$.

LACTVAL : Longueur actuelle de la VALEUR du domaine.

Longueur : 1 byte.

Valeurs possibles : $0 \leq \text{LACTVAL} \leq \text{LMAXVAL}$.

VALEUR : Valeur du domaine.

Longueur : LACTVAL

Valeurs possibles : suite de caracteres ASCII de code 0 .. 255.

NOMDOM : Nom du domaine.

Longueur : LDOM

Valeurs possibles : Suite de caracteres (A..Z).

TOUTES LES VARIABLES QUI SUIVENT SONT UTILISEES POUR
PASSER DES PARAMETRES AU PROGRAMME DE GESTION DE LA B.D. ET
RECEVOIR UNE REPONSE.

- CAT# Variable alphanumerique utilisee exclusivement
 pour communiquer a la primitive "LOADBD", le nom
 d'une base de donnees.
- REL# Variable alphanumerique utilisee , dans presque
 toutes les primitives (Exepte BEGIN et END) pour
 communiquer le nom d'une relation.
- DOM# Variable alphanumerique utilisee pour communiquer
 aux primitives "MAJDOM" et "READOM", le nom d'un
 domaine de la relation dont le nom est mentionne
 dans la variable REL#.
- ID Variable entiere , indique la cle de recherche a
 utiliser pour le START.
- NREC Variable entiere , utilisee pour communiquer aux
 primitives "READ" et "WRITE" un numero
 identifiant une ligne de la relation.
- IO# Variable alphanumerique, destinee a indiquer le
 type d'ouverture de fichier lors de l'appel a
 la primitive "OPEN". Les seules valeurs reconnues
 sont "INP", "OUT", "EXT", "I/O".

- VALEUR# Variable alphanumerique destinee a passer ou recevoir la valeur d'un domaine d'un n-uplet de l'extension d'une relation.
- ACTION# Variable alphanumerique destinee a informer le programme de gestion de la base de donnees de la primitive a exectuter.
- FL Variable entiere dont la valeur informe l'utilisateur sur le fonctionnement correct des primitives. La valeur 0 indique toujours un fonctionnement normal tandis que une valeur superieure a 0 possede une signification dependante de la primitive executee.

TOUTES LES VARIABLES QUI SUIVENT SONT RESERVEES
EXCLUSIVEMENT POUR LA GESTION DE LA BASE DE DONNEES.

- BDREL\$ Variable alphanumerique destinee a memoriser le
 nom d'une relation, en vue d'eviter de trop
 frequentes recherches dans la table des
 descripteurs. Elle est initialisee a la valeur
 courante de la variable REL\$ de
- BDREL Si BDREL\$="" alors BDREL a une valeur quelconque
 sinon BDT(BDREL) pointe vers le
 descripteur de la relation BDREL\$
- BDDOM\$ Variable alphanumerique destinee a memoriser le
 nom d'un domaine de la relation dont le nom est
 dans la variable BDREL\$, en vue d'eviter des
 recherches trop frequentes du descripteur du
 domaine lors de l'emploi des primitives "READOM"
 et "MAJDOM"
- BDDOM si BDDOM\$="" alors BDDOM a une valeur quelconque
 sinon BDT(BDDOM) pointe vers le
 descripteur du domaine BDDOM\$ de
 la relation BDREL\$.
- BDPDOM1 Variable entiere intermediaire.
 Si BDREL\$="" alors BDPDOM1 a une valeur quelconque

sinon BDT(BDPDOM1) pointe vers le
premier descripteur des domaines
de la relation BDREL#.

BDPDOM2 Variable entiere intermediaire.
Si BDREL#="" alors BDPDOM2 a une valeur quelconque
sinon BDT(BDPDOM2) pointe vers le
dernier descripteur des domaines
de la relation BDREL#.

BDACTREL Variable entiere. Elle indique le nombre de
descripteurs de relations enregistres dans
BDE#().

BDACTDEG Variable entiere. Elle indique le degre de la
relation BDREL#.

BDBUFRE Variable entiere. Elle designe le numero de
buffer perpetuellement reserve pour la gestion
de la base de donnees. Cette variable est
initialisee a un lors de l'utilisation de la
primitive "LOADBD".

TBUFFER#() Cette table alphanumerique indique les buffers
reserves.

Si TBUFFER#(i)<>"" alors le buffer numero (i) est
reserve par la gestion de la base de donnees.

Si TBUFFER\$(i) = "" alors le buffer numero (i) est libre.

Les buffers sont alloues et desalloues respectivement par les primitives "OPEN" et "CLOSE".

- BDBUF Variable entiere intermediaire, contenant un numero de buffer reserve.
- BDBUF1 Variable entiere intermediaire, contenant un numero de buffer reserve pour un fichier d'index.
- BDBUF2 Variable entiere intermediaire, contenant un numero de buffer reserve pour un fichier principal
- BDTIO\$() Cette table alphanumerique est destinee a contenir le type d'ouverture de fichier choisi lors de l'appel de la primitive "OPEN".
- BDTYPE\$ Variable alphanumerique intermediaire.
Si BDREL\$="" alors BDTYPE\$ a une valeur quelconque
sinon BDTYPE\$ indique le type de la relation BDREL\$.
- Cette variable est initialisee des l'utilisation d'une des primitives de la gestion de la B.D. (Exepte "BEGIN" et "END").
- BDLONG() Table numerique.

Si (i) refere a un buffer reserve pour un fichier principal alors BDLONG(i) indique la longueur du buffer numero (i).

BDZONE#() Tableau alphanumerique.

Si BDLONG(i)<>0 alors BDZONE#(i,j) represente la j eme zone du buffer numero (i).

BDZONE#(i,0..BDACTDEG) constitue l'entierete du buffer.

BDINDEX#() Tableau alphanumerique.

Si (i) refere a un buffer reserve pour un fichier principal & (j) un numero de cle de recherche alors BDINDEX#(i,j) indique les zones du buffer qui forment la j eme cle de recherche.

Exemple :	soit	R(A	,	B	,	C)
	longueur		3		2		4	
	num. appartenance		1		2		1	

La premiere cle de recherche est formee des domaines A et C, la deuxieme du domaine B.

Decoupe du buffer (i) reserve au fichier principal.

```

+---I---+---+---I---+---I---+---+---+---+
!   !           !           ! <- 4 bytes -> !
+---I---+---+---I---+---I---+---+---+

```

BDZONE#(i,0) BDZONE#(i,2)

 BDZONE#(i,1) BDZONE#(i,3)

BDZONE#(i,0..3) constituent le buffer.

BDINDEX\$(i,1)="13" (code ASCII 1 et 3)

BDINDEX\$(i,2)="2" (code ASCII 2)

BDINDEX\$(i,3..5)=""

BDTRANS Variable Booleenne.

VRAI --> une transaction est en cours.

FAUX --> pas de transaction en cours.

Cette variable est initialisee a "vrai" par la primitive BEGIN et a "faux" par la primitive "END".

BDTRANS\$(i) Table alphanumerique destinee a memoriser ,dans le cas des relations de type Indexe-Relatif ou Indexe-sequentiel, le nom du fichier principal pour lequel tous les fichiers d'indexs ont ete sauves avant modification.

Le nom des fichiers sont deduits des noms de relation et nous trouverons dans cette table le nom complet du fichier, extension inclue.

Ex: Si R(A,B,C,...) est une relation de type

IDX alors le fichier principal porte le

le nom "R.IDX" et les fichiers

d'index R.IDi (i=numero de cle

de recherche).

BDJOURN Si une transaction est en cours (BDTRANS=vrai) alors cette variable entiere indique le nombre

d'enregistrements dans le journal.

- BDNREC** Variable entiere , interediare, destinee a memoriser un numero d'enregistrement.
- BDNREC(i)** Table d'entiers.
Si le buffer (i) est assigne , par la gestion de la B.D. a un fichier d'index ou un fichier principal
alors BDNREC(i) pointe vers un enregistrement de ce fichier.
- BDLAST(i)** Table d'entier.
Si le buffer numero (i) est assigne , par la gestion de la B.D. , a un fichier d'index.
alors BDLAST(i) pointe vers le dernier enregistrement de ce fichier.
- BDINDIR** Variable entiere, intermediaire. Elle pointe vers l'enregistrement du fichier principal trouve par un START.
- BDFICHPRINC#** Variable alphanumerique, intermediaire. Elle est destinee a memoriser le nom complet du fichier principal de la relation dont le nom est dans BDREL#.

BDKEY(i) Table entiere.

Si le buffer (i) est assigne par la gestion de
la B.D. a un fichier d'index

alors si BDKEY(i)<>0

alors il indique le numero de la cle de
recherche dernierement utilisee par
un START et que le fichier de nom :
BDREL\$+".ID"+ car.ASCII de BDKEY(i)
est ouvert et assigne au buffer(i)
sinon le buffer(i) a ete reserve pour un
fichier d'index mais celui-ci n'est
pas encore ouvert.

LES VARIABLES QUI SUIVENT SONT EXCLUSIVEMENT UTILISEES PAR LA GESTION DE LA BASE DE DONNEES POUR ACCEDER AU CONTENU DES FICHIERS D'INDEXS. DES SOUS-ROUTINES SE CHARGE DU "BLOCAGE" ET "DEBLOCAGE" DES INFORMATIONS LUES DANS CES FICHIERS EN VUE D'ACCELERER LE TRAITEMENT.

BDBCK Cette variable entiere , indique le facteur de blocage a utiliser pour la lecture et l'écriture dans tous les fichiers d'indexs.

BDLOGREC Variable entiere, intermediaire.
Elle indique le numero d'enregistrement logique a lire ou a écrire.

BDRECLU Variable entiere, intermediaire.
Contenu de l'enregistrement logique dernièrement lu par la gestion de la B.D.

BDRECECR Variable entiere, intermediaire.
Contenu de l'enregistrement logique a écrire.

BDRECPHYS() Table d'entiers.

Si (i) est un numero de buffer reserve pour un fichier d'index et si BDRECPHYS(i) <> 0 alors BDRECPHYS(i) indique le numero de l'enregistrement physique memorise dans le buffer (i). Celui -ci est constitue des zones :

BDZONE#(i,0)... BDZONE#(i,BDBCK)

Exemple : Si BDBCK=8 ,BDRECPHYS=2 et le buffer i reserve pour un fichier d'index R.ID1 :

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+...
!   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+...
I  1  2  3  4  5  6  7  8I  9  10 11 12  13 14 15 16I17  ....
I                                     I                                     I
I  prem. enreg. Phys.      I second enreg. phys.      I

```

13 ieme enregistrement logique
(BDLOGREC=13)

LES VARIABLES QUI SUIVENT SONT EXCLUSIVEMENT UTILISEES PAR LA GESTION DE LA BASE DE DONNEES POUR COMMUNIQUER CERTAINS PARAMETRES AU PROGRAMME CHARGE D'EFFECTUER LA RECHERCHE DICHOTOMIQUE DANS LE FICHIER PRINCIPAL VIA UN FICHIER D'INDEX.

BDDEBUT Variable entiere, intermediaire.
Elle indique le numero du premier enregistrement du fichier d'index.

BDFINIDX Variable entiere, intermediaire.
Elle indique le numero du dernier enregistrement d'un fichier d'index.

BDCLE# Variable alphanumerique, intermediaire.
Elle contient la vakeur d'une cle de recherche.

CONTRAINTES PARTICULIERES A RESPECTER LORS DE LA MODIFICATION DE LA VALEUR DU FACTEUR DE BLOCAGE.

- 1) $0 < BDBCK < 124$
- 2) le nombre de colonnes du tableau $BDZONE\#(i,j)$ doit etre superieur ou egal a $BDBCK$. ($j \geq BDBCK$).

4.1.5.3. Transfert des donnees

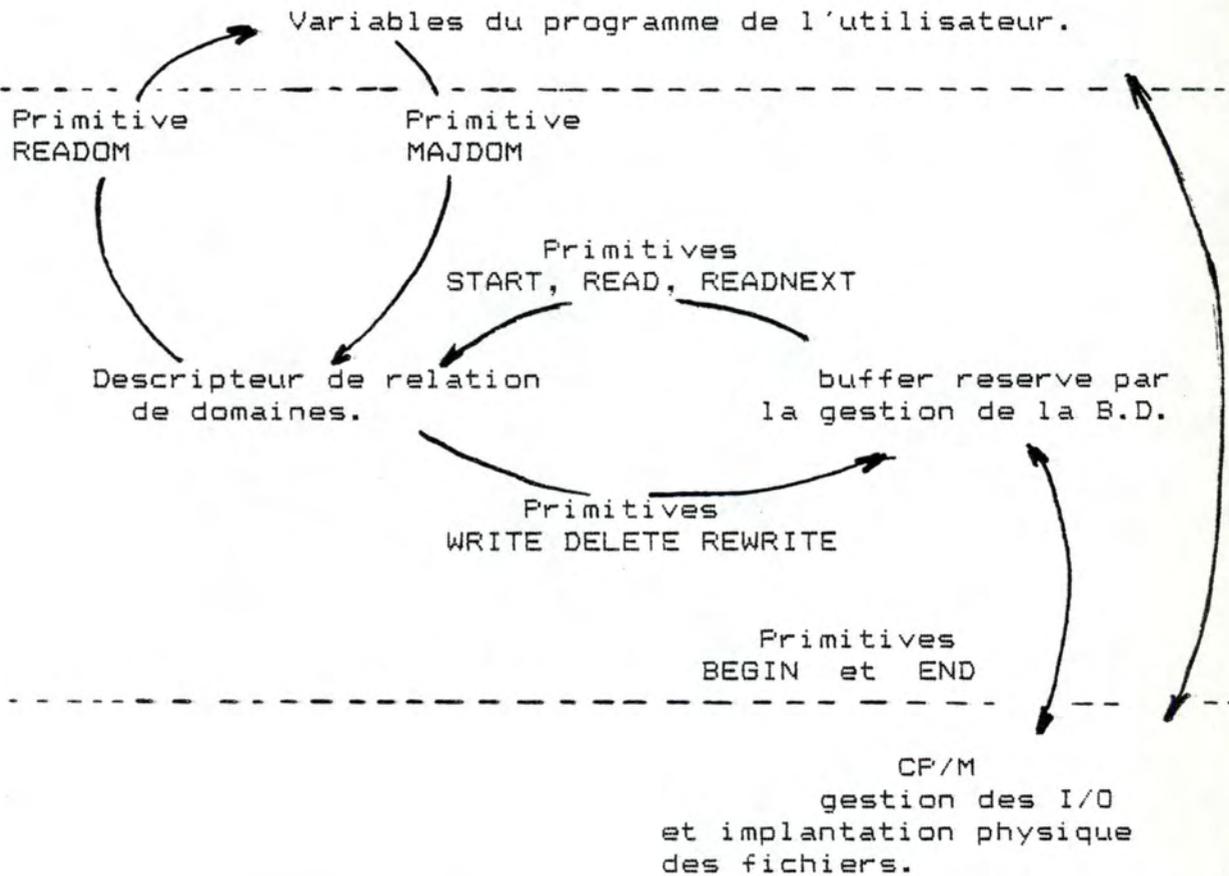
Les primitives READOM et MAJDOM agissent directement sur les descripteurs de domaines d'une relation. (Table BDE#).

Les primitives READ , READNEXT et START accedent aux fichiers et transferent les informations trouvees des buffers d'entree/sortie vers les zones nommes "VALEUR" des descripteurs des domaines d'une relations.

Les primitives REWRITE, DELETE, WRITE transferent la valeur courante des zones nommes "VALEUR" des descripteurs des domaines d'une relation vers les fichiers en passabt par les buffers d'entree/sortie.

L'utilisateur a toujours la possibilite d'acceder a des fichiers sans passer par l'intermediaire des programmes de la gestion de la base de donnees.

 SCHEMA DES TRANSFERTS DE DONNEES ENTRE LES PROGRAMMES DU
 PROGRAMMEURS, LA GESTION DE LA BASE DE DONNEES, LES BUFFERS
 ET LES FICHIERS.



DESCRIPTION DETAILLEE DE CERTAINS TRANSFERTS DE DONNEES.

DESCRIPTEURS DE LA RELATION R(A,B,C)												
BDT	← table →								BDE#			
! 0 !	<0>	! 2 !	! 3 !	! 1 !	!	16	!	B	D	!		
! 1 !	<1>	! 1 !	I	D	X	!	4	!	3	!	4	!
! 0 !	<2>	! 1 !	!	1	!	3	!	2	!	a	a	!
! 0 !	<3>	! 1 !	!	0	!	4	!	4	!	t	o	!
! 2 !	<4>	! 1 !	!	1	!	2	!	1	!	X	!	C
! 3 !	<5>	!!										
! 4 !	<6>	!!										
! 0 !	<7>	!!										

BDZONE#(i,*)	!	.	!	a	a	!	t	o	t	o	!	X	!
--------------	---	---	---	---	---	---	---	---	---	---	---	---	---

BDZONE#(i,0) BDZONE#(i,1) BDZONE#(i,2) BDZONE#(i,3)
 Buffer reserve et decoupe (lors de l'OPEN) en differentes zones. Chacune correspondant a un domaine bien specifique d'une meme relation.

4.1.5.4 Description des algorithmes.

Primitives d'accès a un domaine

Nous partirons d'une structure initiale des descripteurs correspondant a un catalogue nomme "BD" contenant la description d'une relation R(A,B,C) de type IDX , chaque domaine ayant une longueur maximale de 3 caracteres. Nous supposerons pour nous faciliter la tache que le programmeur a prevu la creation d'un maximum de 3 relations avec un maximum de 4 domaines definissable pour chacune d'entre elles.

Nous definirons une seule cle de recherche sur les domaines "A" et "C".

Structure initiale.

tableau BDT		BDE#																		
+	+	+	+	+	+	+	+	+												
!	0	!	<0>	!	2	!	3	!	1	!	16	!	B	!	D	!				
!	1	!	<1>	!	1	!	I	!	D	!	X	!	4	!	3	!	4	!	R	!
!	0	!	<2>	!	1	!	1	!	3	!	0	!	A	!						
!	0	!	<3>	!	1	!	0	!	3	!	0	!	B	!						
!	2	!	<4>	!	1	!	1	!	3	!	0	!	C	!						
!	3	!	<5>	!!																
!	4	!	<6>	!!																
!	0	!	<7>	!!																

Nous trouvons dans BDE#(BDT(0)) le descripteur de catalogue ,dans BDE#(BDT(1)) le descripteur de la relation "R" , dans BDE#(BDT(4)) le descripteur du premier domaine "A" de la relation "R" ... et en BDE#(BDT(6)) le descripteur du dernier domaine "C" de la relation "R".

Nom de la primitive : MAJDOM.

Supposons que le programmeur desire initialiser le domaine "A" de la relation "R" avec la valeur "aaa", le domaine "B" avec la valeur "bbb", le domaine "C" avec la valeur "ccc", en vue d'ajouter une nouvelle ligne a une relation. Il programmera la pseudo-sequence d'exécution des primitives suivante :

```
MAJDOM( "R","A","aaa")
MAJDOM( "R","B","bbb")
MAJDOM( "R","C","ccc")
WRITE ( "R" )
```

qui aura pour effet de modifier le contenu initial des descripteurs des domaines "A", "B" et "C" comme suit :

tableau BDT				BDE#			
+	---	+	---	+	---	+	---
! 0 !	<0>	! 2 !	3 !	! 1 !	16	! B	D !
! 1 !	<1>	! 1 !	I	D	X	! 4 !	3 !
! 0 !	<2>	! 1 !	1 !	3 !	3 !	a	a
! 0 !	<3>	! 1 !	0 !	3 !	3 !	b	b
! 2 !	<4>	! 1 !	1 !	3 !	3 !	c	c
! 3 !	<5>	!!					
! 4 !	<6>	!!					
! 0 !	<7>	!!					

Les seules primitives qui peuvent modifier la valeur d'un domaine d'une relation et par consequent les descripteurs correspondant sont : MAJDOM, START, READNEXT et READ.

Nom de la primitive : READOM.

Supposons maintenant que le programmeur desire lire la valeur du domaine de nom "A" de la relation "R".

Il utilisera la primitive READOM et recueillera dans une variable de programme BASIC la valeur de la zone "VALEUR" du descripteur du domaine "A" de la relation "R".

Les descripteurs ne sont pas modifies.

 Primitives d'accès à la base de données.

Nous décrirons les modifications apportées à la structure des fichiers physiques correspondant à une relation $R(A,B,C)$. Pour les fichiers indexé-sequentiels et indexé-relatifs nous définirons une seule cle de recherche sur le domaine "A" et "C" dans le but de simplifier les explications. Nous rappelons que l'on peut définir au plus cinq cles de recherche non nécessairement distinctes.

 E X E M P L E D E B A S E

soit la relation $R(A , B , C)$ et l'extension suivante :

"aaa"	,"bbb"	,"ccc"
"zzz"	,"123"	,"456"
"aaa"	,"ccc"	,"aaa"
"eee"	," "	,"eee"

pour un fichier indexe sequentiel. (IDX)

Voici le contenu initial des differents fichiers physiques.

Le nombre entre crochet indique le numero d'enregistrement du fichier relatif-basic.

R.ID1	R.IDX (noms des fichiers physiques)										
5	<1>	4									
4	<2>	255	a	a	a	b	b	b	c	c	c
2	<3>	255	z	z	z	1	2	3	4	5	6
5	<4>	255	a	a	a	c	c	c	a	a	a
3	<5>	255	e	e	e				e	e	e
0	<6>	0									
.
.

fichier index fichier principal.

Pour un fichier indexe relatif. (IDR)

R.ID1	R.IDR (noms des fichiers physiques)										
5	<1>	255	a	a	a	b	b	b	c	c	c
3	<2>	255	z	z	z	1	2	3	4	5	6
1	<3>	255	a	a	a	c	c	c	a	a	a
4	<4>	255	e	e	e				e	e	e
2	<5>	0									
0	<6>	0									
.
.

fichier index fichier principal.

Nom de la primitive LOADBD.

Cette primitive initialise les tables BDE#() et BDT()
(voir 4.1.5.2) avec le contenu de deux fichiers contenant
la description d'une base de donnees.(voir 4.1.5.1).

Nom de la primitive : OPEN

Cette primitive recherche un(des) buffer(s) libre(s)
,le(s) reserve , decoupe le buffer reserve au fichier
principal en fonction de la description d'une relation ,
et ouvre ce fichier. Elle effectue ensuite les
initialisations necessaires a l'utilisation d'autres
primitives d'accès a la base de donnees.

Nom de la primitive : CLOSE

Cette primitive libere le(s) buffer(s) reserve(s) a
L'OPEN. et ferme le(s) fichier(s) correspondant a une
relation.

Nom de la primitive : START

Si la valeur des domaines A et C , au moment de l'appel a la primitive START , sont respectivement "aaa" et "bbb" et que l'utilisateur desire effectuer la recherche sur la premiere cle de recherche. Il obtiendra respectivement pour un fichier IDX et IDR , le deuxieme et le premier enregistrement. Il accedera aux valeurs des differents domaines en employant la primitive READOM. La lecture de la valeur des domaines "A" "B" et "C" par cette primitive fournira respectivement les chaines de caracteres "aaa" , "bbb" et "ccc".

Nom de la primitive READNEXT

Suite de l'exemple precedent , l'emploi de la primitive READNEXT permettra d'accéder a l'enregistrement suivant du fichier principal trie sur la cle de recherche. La lecture de la valeur des domaines "A" , "B" et "C" par la primitive READOM fournira respectivement les chaines de caracteres "eee" , " " , "eee". Une variable indiquera si l'on a atteint la fin de fichier.

Nom de la primitive WRITE.

Supposons que l'on desire ajouter une ligne a l'extension de la relation R soit "ccc","ccc","ccc". Il faut initialiser les domaines d'une future ligne de la relation en employant la primitive MAJDOM avant d'appeler la primitive WRITE.

Pour un fichier indexe relatif. (IDR)

Pour les fichiers indexe-relatifs il est necessaire d'indiquer un numero identifiant par exemple le numero 6. Voici le contenu des differents fichiers physiques apres l'ajout d'une nouvelle ligne. L'operation implique l'insertion d'un nouveau enregistrement dans le fichier d'index.

R. ID1			R. IDR (noms des fichiers physiques)														
!	6	!	<1>	!	255!	a	a	a	!	b	b	b	!	c	c	c	!
!	3	!	<2>	!	255!	z	z	z	!	1	2	3	!	4	5	6	!
!	1	!	<3>	!	255!	a	a	a	!	c	c	c	!	a	a	a	!
!	6	!	<4>	!	255!	e	e	e	!				!	e	e	e	!
!	4	!	<5>	!	0	!			!				!				!
!	2	!	<6>	!	255!	c	c	c	!	c	c	c	!	c	c	c	!
!	0	!	<7>	!	0	!			!				!				!
!	.	!	.	!	.	!			!				!				!
+-----+			+-----+														
fichier index			fichier principal.														

pour un fichier relatif (REL).

On retrouvera le meme contenu que le fichier principal R.IDR (qui porte un autre nom (R.REL)) , sans fichiers d'indexs.

Il faut egalement mentionner un numero identifiant.

Remarque commune aux deux types de fichiers IDR et REL :
 La primitive refuse de reecrire sur un enregistrement deja
 existant. Dans cet exemple il est donc impossible de
 modifier les lignes de 1 a 4.

R.REL (noms du fichier physique)

<1>	!255!	a	a	a	!	b	b	b	!	c	c	c	!
<2>	!255!	z	z	z	!	1	2	3	!	4	5	6	!
<3>	!255!	a	a	a	!	c	c	c	!	a	a	a	!
<4>	!255!	e	e	e	!				!	e	e	e	!
<5>	! 0 !				!				!				!
<6>	!255!	c	c	c	!	c	c	c	!	c	c	c	!
<7>	! 0 !				!				!				!

Nom de la primitive : REWRITE

Supposons que l'on desire modifier une ligne de l'extension de la relation R. Par exemple remplacer la troisieme ligne ("aaa","ccc","aaa") par la ligne "xxx","xxx","xxx". Il faudra initialiser les domaines de la future ligne de la relation en employant la primitive MAJDOM avant d'appeler la primitive REWRITE.

Tous les fichiers indexes (IDR et IDX) necessitent une lecture prealable par une primitive START ou READNEXT. Nous supposerons pour les besoin de notre exemple que l'on a deja effectue un positionnement sur la troisieme ligne de la relation par un START ou un READNEXT. Une caracteristique interressante de la primitive REWRITE est qu'elle ne modifie le contenu des fichiers indexes que si cela s'avere necessaire.

Exemple : Si nous avons decide de remplacer la troisieme ligne de cette relation par "aaa" , "xxx" , "aaa" , le programme sous jacent a cette primitive aurait simplement replacer le contenu du troisieme enregistrement sans toucher au contenu du fichier d'index.

Pour un fichier indexe relatif. (IDR)

Voici le contenu des differents fichiers physiques apres l'operation. Elle implique l'insertion et l'effacement d'un enregistrement dans le fichier d'index.

R.ID1	R.IDR (noms des fichiers physiques)													
5	<1>	!255!	a	a	a	!	b	b	b	!	c	c	c	!
1	<2>	!255!	z	z	z	!	1	2	3	!	4	5	6	!
4	<3>	!255!	x	x	x	!	x	x	x	!	x	x	x	!
3	<4>	!255!	e	e	e	!				!	e	e	e	!
2	<5>	! 0 !				!				!				!
0	<6>	! 0 !				!				!				!

fichier index fichier principal.

pour un fichier relatif (REL).

On retrouvera le meme contenu que le fichier principal R.IDR (qui porte un autre nom (R.REL)) , sans fichiers d'indexs.

Il faut mentionner le numero identifiant la ligne a reecrire (par Exemple la ligne numero 3).

R.REL (nom du fichier physique)													
<1>	!255!	a	a	a	!	b	b	b	!	c	c	c	!
<2>	!255!	z	z	z	!	1	2	3	!	4	5	6	!
<3>	!255!	x	x	x	!	x	x	x	!	x	x	x	!
<4>	!255!	e	e	e	!				!	e	e	e	!
<5>	! 0 !				!				!				!
<6>	! 0 !				!				!				!

Nom de la primitive : DELETE.

Supposons que l'on desire effacer une ligne de l'extension de la relation R. Par exemple effacer la troisième ligne ("aaa","ccc","aaa").

Tous les fichiers indexés (IDR et IDX) nécessitent une lecture préalable par une primitive START ou READNEXT. Nous supposons pour les besoins de notre exemple que l'on a déjà effectué un positionnement sur la troisième ligne de la relation par un START ou un READNEXT.

Nom de la primitive : READ.

Cette primitive n'est utilisee que pour le fichier sequentiel (SEQ) et le fichier relatif (REL).
Pour un fichier sequentiel (SEQ)

Cette primitive permet de lire l'enregistrement suivant le dernier lu a partir du debut du fichier.

On retrouvera le meme contenu que le fichier principal R.IDX (qui porte un autre nom (R.SEQ)) , sans fichiers d'index.

R.SEQ (nom du fichier physique)

<1>	!	5	!								!			
<2>	!	255!	a	a	a	!	b	b	b	!	c	c	c	!
<3>	!	255!	z	z	z	!	1	2	3	!	4	5	6	!
<4>	!	0	!	a	a	a	!	c	c	c	!	a	a	a
<5>	!	255!	e	e	e	!				!	e	e	e	!
<6>	!	0	!											!

Pour un fichier relatif (REL) et indexe-relatif (IDR).

Cette primitive permet de lire l'enregistrement dont le numero est egal au numero identifiant.

4.2 Gestion d'écran.

4.2.0 Introduction.

Le sous-système de gestion d'écran développé est assez rudimentaire.

Nuançons cette remarque :

les primitives d'exploitation des masques de saisie définis ont été soigneusement élaborées, elles contiennent toutes les validations nécessaires; mais les primitives de description d'une grille de saisie ne devaient être utilisées qu'une fois par nous pour définir une fois pour toutes les masques de saisie utilisés dans nos différents programmes; c'est pourquoi nous n'avons pas jugé nécessaire de prévoir toutes les validations utiles d'autant plus que cela nous a permis d'alléger des programmes déjà très grands afin d'essayer d'éviter au maximum les chaînages de programmes qui alèrent fortement les performances du logiciel.

Néanmoins, il peut être intéressant pour l'utilisateur de dessiner lui-même ses écrans et d'en changer de temps à autre le format, une manière comme une autre de sortir un petit peu de la routine.

Le sous-système "gestion d'écran" sera amélioré en conséquence ultérieurement. (cfr. chap. 6)

4.2.1 Concepts de base et nomenclature.

Un écran ou encore grille ou masque de saisie est un ensemble de zones. On donne un nom à cet écran.

Ces zones sont de deux types :

- zones protégées : zones dans lesquelles l'utilisateur ne peut pas écrire,
- zone non protégées : l'utilisateur peut écrire dans ces zones.

Les zones du second type sont bien entendu destinées à saisir les données, alors que les zones protégées sont destinées à contenir des commentaires qui rappelleront à l'utilisateur quel est le contenu attendu des différentes zones non protégées.

Illustrons ces concepts sur un exemple :

"NOM DU PATIENT : - - - - - " pourrait être une partie d'un écran.

La deuxième zone matérialisée par des pointillés est destinée à saisir la valeur de la variable nompatient. La première zone lui sert d'en-tête : elle rappelle à l'utilisateur, qu'à cet endroit de l'écran on doit introduire le nom du patient.

Examinons les différents paramètres qui caractérisent une zone.

Nous ne reviendrons pas sur le caractère protégé ou non d'une zone.

Une zone est caractérisée par sa position et sa longueur. Sa position sera donnée par les coordonnées du début de la zone (numéro de colonne, numéro de ligne).

Pour une zone protégée, la longueur est la longueur de l'intitulé de la zone (ex. : "NOMDUPATIENT :" est de longueur 15). Pour une zone non protégée, il s'agit de la longueur maximale que peut prendre une valeur saisie dans la zone.

Nous nous intéressons maintenant aux zones non protégées. Le lecteur se rendra vite compte que les paramètres énoncés ci-après n'offrent aucun intérêt pour les zones protégées.

Nous allons caractériser la valeur qui pourra être contenue dans la zone.

La valeur peut être de type alphanumérique, numérique ou du type date.

La zone peut être obligatoire ou non : lors de la saisie l'utilisateur devra obligatoirement mettre une valeur non vide dans la zone.

Pour que la donnée saisie dans la zone soit valide il faut parfois que la valeur soit incluse dans un intervalle de valeurs. On peut donner, pour une zone la valeur minimum et la valeur maximum que peut prendre la donnée.

A une zone sont encore associés deux nombres qui serviront de pointeurs : l'ensemble des pointeurs de toutes

les zones permet de constituer le parcours des zones de l'écran en avant et en arrière.

Il faut bien entendu que nous puissions identifier une zone. Chaque zone a un nom et un numéro.

Il est souvent intéressant de pouvoir afficher des morceaux de l'écran. Nous appellerons ces morceaux : 'SECTIONS'. Il faut donc préciser pour chaque zone, la section à laquelle elle appartient.

Enfin, à chaque zone est associé un texte d'aide qui contiendra tous les renseignements nécessaires concernant le contenu attendu de la zone.

4.2.2 Fonctions de base.

Nous allons simplement énumérer les différentes fonctions que nous avons envisagées et nous essayerons de mettre en évidence leur utilité.

Chacune de ces fonctions ou primitives sera explicitée ultérieurement. Ce paragraphe est destiné à donner une bonne vue d'ensemble du système de gestion d'écran.

Il faut bien évidemment des primitives de description d'un masque de saisie, à savoir créer et modifier une description sous base du nom de l'écran à générer ou à modifier.

Il est utile, à ce niveau, de pouvoir éditer un fichier contenant un récapitulatif de la description d'un écran. Il peut aussi être intéressant de pouvoir éditer une image du masque de saisie.

Examinons les primitives d'exploitation.

L'utilisateur doit pouvoir écrire dans une zone, demander l'affichage d'une donnée dans une zone. Il faut aussi pouvoir initialiser l'écran : effacer l'écran puis afficher un masque vierge après en avoir chargé sa description.

A chaque zone, nous avons associé un texte d'aide : on doit pouvoir accéder à ce texte et ensuite restaurer l'écran dans l'état précédent la demande d'aide. Il faut donc, au préalable, sauver le contenu de l'écran.

Nous permettons de définir les caractéristiques de la zone : obligatoire, numérique,...

Différentes primitives vérifieront la validité des données par rapport à ces paramètres.

Nous allons examiner ces primitives une à une.

Le schéma de présentation est le suivant :

nom : nom de la fonction.

entrée : les paramètres d'entrée de la fonction.

sortie : les paramètres de sortie de la fonction.

mode d'appel : le basic utilisé ne permet pas la définition et l'appel de fonction. Il faudrait après avoir initialisé les différents paramètres, effectuer un branchement à un numéro de ligne.

Avec ce système, l'utilisation de la gestion d'écran est très lourde; les programmes sont illisibles.

Nous avons donc simulé l'appel de fonction : le nom de la primitive à exécuter est passé, avec les autres paramètres à un programme qui effectuera le branchement adéquat grâce au nom de la fonction.

effet : dans ce point, nous décrirons ce que fait (°) la fonction primitive en fonction des paramètres d'entrée et des paramètres de sortie.

Voici la liste des variables utilisées :

ecran\$: contient le nom de l'écran considéré
 nozone : numéro de la zone courante
 DESC\$: table contenant la description de l'écran courant
 I\$: première entrée libre dans la table DESC
 fction\$: nom de la primitive à exécuter
 coderet : code de retour.

Examinons le contenu de DESC\$.

DESC\$ est un tableau à deux dimensions.

Chaque ligne est le descripteur d'une zone.

DESC\$ (,1) = numéro de la zone

DESC\$ (,2) = nom de la zone

DESC\$ (,3) = caractère protégé ou non de la zone

DESC\$ (,4) = numéro de colonne du début de la zone

DESC\$ (,5) = numéro de ligne du début de la zone

DESC\$ (,6) = longueur max. de la zone

DESC\$ (,7) = libellé des zones protégées ou en cours de saisie le contenu des zones non protégées

(°) Pour connaître le "comment" consulter les paragraphes consacrés à l'implémentation des primitives.

DESC \$ (,8) = caractère obligatoire de la zone
 DESC \$ (,9) = type de la zone (numérique, date)
 DESC \$ (,10) = numéro de la section à laquelle appartient la zone
 DESC \$ (,11) = valeur minimum
 DESC \$ (,12) = valeur maximum
 DESC \$ (,13) et DESC (,14) = pointeurs.

4.2.3 Les primitives d'exploitation.

Toutes ces primitives sont regroupées dans le fichier "ACCESECR.BAS".

Pour pouvoir faire appel à ces primitives, il suffit d'assigner le nom de la fonction à la variable FCTION\$, d'initialiser les différents paramètres et d'effectuer un branchement à la routine d'aiguillage vers les différentes fonctions (gosub 10014).

Mise à blanc de l'écran.

nom : CLEAR
 entrée : ∅
 sortie : ∅
 effet : met à blanc l'écran vidéo.

Initialisation de l'écran.

nom : INI
 entrée : écran\$: nom de l'écran ; section : numéro de la section de l'écran à afficher
 sortie : /

effet : affiche la grille vide ou la section de l'écran correspondant à la description contenue dans "écran\$ ".seq.
 Si la section n'est pas précisée, tout l'écran est affiché.
 Si le fichier contenant la description de écran n'existe pas, le message "file not found" est affiché.
 Le programme s'arrête.

Production de message d'erreur.

nom : ERREUR
 entrée : nozone : numéro de la zone pour laquelle on désire afficher un message d'erreur
 mess\$: message d'erreur
 sortie : coderet : code d'erreur
 effet : - reinitialise la zone considérée
 - le message est affiché dans le bas de l'écran
 - avant cela on vérifie que l'on affiche bien un message pour une zone non protégée; dans ce cas coderet= 0, sinon coderet = 1.

Vérification du caractère numérique d'une zone.

nom : NUM
 entrée : zone\$, valeur à examiner
 sortie : coderet
 effet : vérifie que la zone contient bien une valeur numérique \Rightarrow coderet = 0
 sinon coderet = 1.

Sauvegarde du contenu d'un écran.

nom : SAVE

entrée : DESC \$ contenant la description de l'écran et
la valeur courante des différentes zones non
protégées
écran \$

sortie : écran \$.seq

effet : le contenu de l'écran et sa description sont co-
piés dans le fichier "écran ".seq.

Restauration d'un écran.

nom : REST

entrée : écran \$
DESC \$
I \$

sortie : DESC \$
écran \$.seq

effet : - affiche l'écran dans l'état précédent le sau-
vetage.
- le fichier contenant la description de l'écran
est nettoyé : le contenu des zones non protégées
est effacé du fichier.

Vérification du remplissage des zones obligatoires.

nom : OBLI

entrée : DESC \$

sortie : le vecteur OBLI

effet : parcourt toutes les zones obligatoires.

Le numéro des zones obligatoires non remplies
est communiqué à l'utilisateur via le vecteur
OBLI.

Ecriture dans une zone de l'écran.

nom : WRITE

entrée : Dom\$, nom de la zone dans laquelle on va afficher une donnée
 valeur\$, valeur à afficher dans la zone

sortie : coderet

effet : teste si l'on affiche bien dans une zone non protégée

Si oui codere t = 0

Si non codere t = 1

la donnée est affichée dans la zone.

Validation d'une date.

nom : DATE

entrée : date\$ date à analyser

sortie : coderet

effet : vérifie la validité de la date

si la date est correcte coderet = 0, si non
 coderet = 1.

Lecture d'une zone de l'écran.

(l'utilisateur écrit dans la zone)

nom : READ

entrée : nozone, numéro de la zone dans laquelle l'utilisateur va écrire.

sortie : coderet

effet : - saisie de la valeur introduite dans la zone avec toutes les validations, suivant le type de la zone (appel à num, obli, date...)
 - si l'utilisateur demande l'écriture dans une zone protégée, codret = 1.

Validation des bornes d'une zone.

nom : BORNE

entrée : nozone, numéro de la zone à valider

sortie : coderet

effet : vérifie que la donnée saisie est compatible avec
les valeurs minimum et maximum qui ont été en-
registrées lors de la saisie description de l'
écran

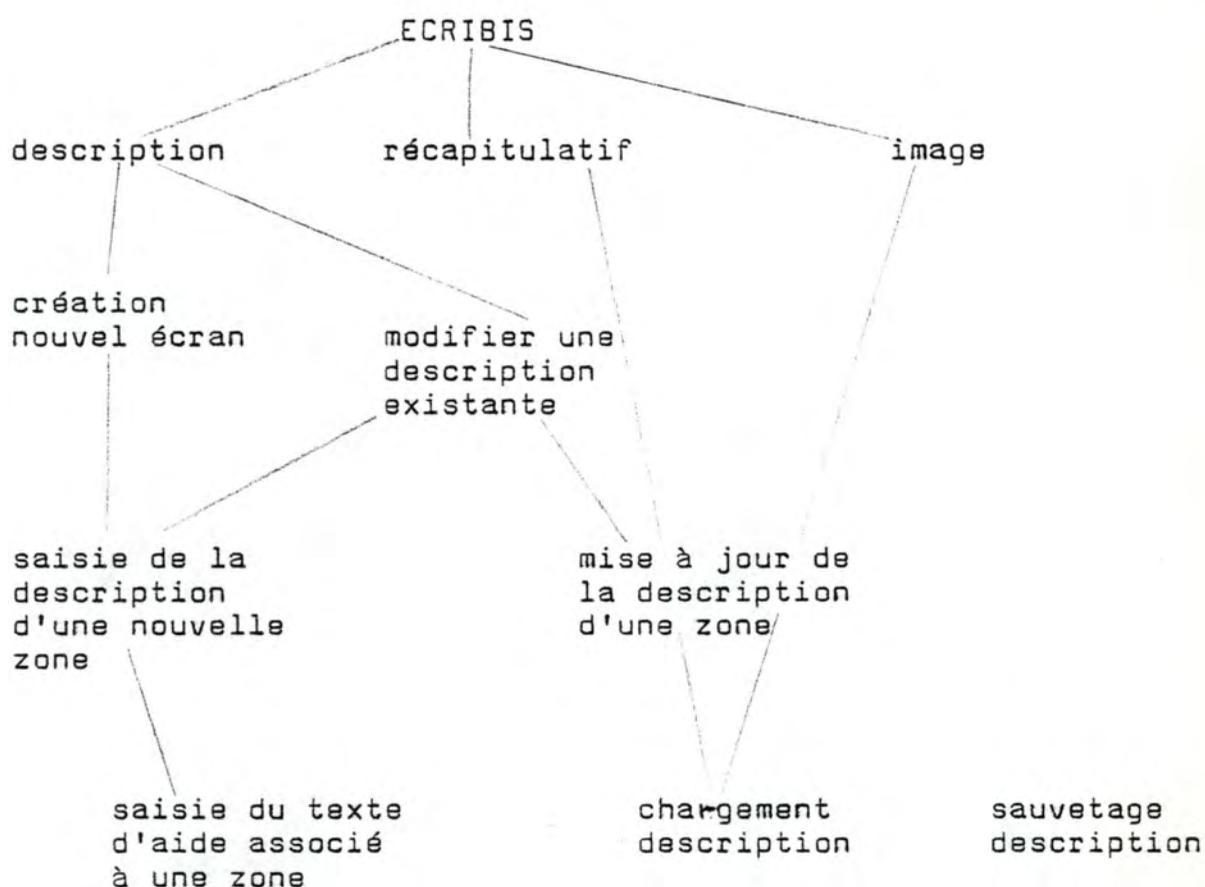
si oui coderet = 0

si non coderet = 1.

4.2.4 Les primitives de description.

Les primitives de description ne sont pas directement utilisables; elles ont été intégrées dans un seul programme : ECRBIS2.BAS

La figure suivante présente la structure statique des appels des différentes routines (primitives).



ECRBIS2

nom : ecrbis2

entrée : ∅

sortie : ∅

effet : - affiche un menu

- en fonction du choix de l'utilisateur, effectue les branchements aux routines adéquates ou stoppe le déroulement du programme.
- Tant que le choix de l'utilisateur n'est pas valide, on itère sur la question du choix de l'utilisateur.

Image.

nom : ∅

entrée : écran# nom de l'écran dont on désire éditer une image

sortie : le fichier "écran# ".ima (°) contenant une grille vide

effet : -chargement de la description de l'écran

- génération d'un fichier ("écran# ".ima) contenant une grille vide
- si le fichier contenant la description de l'écran n'existe pas, le message "file not found" est affiché et le programme s'arrête
- si la description de l'écran n'est pas valide, chevauchement de zones, le programme s'arrête.

(°) "écran# " signifie la valeur de la variable écran .

Récapitulatif.

nom : /

entrée : écran , nom de l'écran dont on désire éditer
la description

sortie : le fichier "écran ".lst contenant un récapitulatif de la description de l'écran

effet : - chargement de la description de l'écran.
- génération du fichier "écran ".lst contenant un résumé de la description de l'écran.
- si le fichier contenant la description de l'écran n'existe pas, le message "file not found" est affiché; le programme s'arrête.

Description.

nom : /

entrée : /

sortie : /

effet : demande à l'utilisateur sous base d'un numéro, s'il désire générer une nouvelle grille ou s'il désire modifier une description existante.
En fonction du choix de l'utilisateur, les branchements adéquats sont effectués.

Création d'un nouvel écran.

nom : /

entrée : écran , nom de l'écran à générer

sortie : le fichier "écran ".seq contenant la description de l'écran et l'indication de la première entrée libre dans la table.

le fichier "help".idx contenant les textes d'aide associés aux différentes zones non protégées.

effet : itération sur le module de saisie de la description d'une zone jusqu'à ce que RII soit égal à 1.

Saisie de la description d'une nouvelle zone.

nom :

entrée : I\$ première entrée libre dans la table des descripteurs de zones.

DESC\$ table contenant les descripteurs de zones
Help.idx

sortie : DESC\$ complétée par le descripteur de la nouvelle zone.

Help.idx mis à jour.

effet :-saisie des différents paramètres de la zone
-attribution automatique d'un numéro à la zone
-pour autant qu'il n'existe pas encore de texte d'aide pour une autre zone portant le même nom, le texte d'aide associé à la zone est saisi.

Remarque : toutes les validations sur les réponses données par l'utilisateur aux différentes questions n'ont pas été envisagées, faute de place mémoire nous avons allégé au maximum les programmes qui ne devaient initialement être utilisés que par nous. C'était le cas pour la gestion d'écran. Toutes ces validations seront ajoutées ultérieurement. Ainsi l'utilisateur doit veiller lui-même au non-chevauchement des zones, à respecter les bornes limites pour les numéros de colonne et de ligne....

Saisie du texte d'aide associé à une zone.

nom : /
 entrée : DESC\$, nozone
 help.idx
 sortie : help.idx
 effet : cherche dans le fichier "help.idx" si un texte
 d'aide a déjà été défini pour la zone.
 Si ce n'est pas le cas, on saisit ce texte.

Modifier une description existante.

nom : /
 entrée : écran\$: nom de l'écran dont on désire modifier
 la description
 sortie : "écran\$.seq, fichier contenant la nouvelle
 description de l'écran.
 effet : - si le fichier contenant la description n'existe
 pas, le message "file not found" est affiché,
 le programme s'arrête
 - le programme itère sur la routine de mise à
 jour de la description d'une zone.

Mise à jour de la description d'une zone.

nom : /
 entrée : - numéro de la zone dont on désire modifier (NO)
 la description
 - DESC\$ - help.idx
 sortie : - DESC\$ et help.idx mis à jour
 effet : si NO est le numéro d'une zone existante, les
 nouveaux paramètres sont saisis.
 On offre la possibilité à l'utilisateur de chan-
 ger le texte de help associé à la zone; sinon

s'il s'agit d'un ajout on saisit les paramètres de la zone et le texte d'aide
 sinon erreur, retour à la routine "modification d'une description existante".

Chargement de la description.

nom :

entrée : écran\$ nom de l'écran dont on doit charger
 la description

sortie : I\$ 1ère entrée libre dans la table
 DESC\$ table des descripteurs de zone.

effet : lit la description de l'écran et la range dans
 I\$ et DESC\$.

Sauvetage de la description.

nom :

entrée : I\$
 DESC\$, écran

sortie : "écran ".seq fichier contenant la description
 de l'écran

effet : écrit I\$ et DESC\$ dans le fichier "écran".seq.

4.2.5 Implémentation.

4.2.5.1 Les descripteurs.

Les descripteurs ont été présentés au point 4.2.2. La description de l'écran, à savoir I\$ la première entrée libre dans la table des descripteurs et la table des descripteurs DESC\$, est rangée dans un fichier séquentiel. Le nom de ce fichier est le nom de l'écran suivi de ".seq".

Rappelons la liste des variables utilisées :

écran \$: nom de l'écran considéré
nozone : numéro de la zone courante
DESC\$: table des descripteurs
I\$: première entrée libre dans la table
fction\$: nom de la primitive à exécuter
coderet : code de retour.

4.2.5.2 Primitives d'exploitation.

Clear.

entrée :

sortie :

réalisation : envoie à l'écran du caractère de contrôle qui permet de l'effacer.

Ini (initialisation de l'écran).

entrée : écran \$, nom de l'écran, section, numéro de la section de l'écran à afficher.

sortie :

Le module charge la description de l'écran (simple lecture du fichier contenant la description).

On parcourt un à un les descripteurs des différentes zones qui composent l'écran.

Si la zone appartient à la section considérée (on teste desc\$ (,10), elle est affichée : s'il s'agit d'une zone protégée on affiche le libellé (desc\$ (,7)), sinon on affiche x tirets où x est la longueur maximale de cette zone (desc\$ (,6)).

Si le numéro de section n'a pas été précisé, on affiche tout l'écran.

ERREUR (production de message d'erreur).

entrée : nozone, numéro de la zone pour laquelle un message d'erreur doit être affiché mess \$, message d'erreur.

sortie : coderet. code d'erreur.

On teste si le nozone correspond bien à une zone non protégée.

Si ce n'est pas le cas coderet prend la valeur 1 et retour au programme qui a appelé ce module.

Sinon : la zone considérée est réinitialisée et le message est affecté dans le bas de l'écran.

Num : (vérification du caractère numérique d'une zone).

entrée : zone \$, valeur à analyser.

sortie : coderet.

La valeur est examinée caractère par caractère (comparaison du code ASCII du caractère aux codes ASCII correspondant à des chiffres).

Dès qu'un caractère non numérique est détecté, coderet prend la valeur 1 et retour au programme appelant :

Save (sauvegarde du contenu de l'écran).

entrée : Desc\$ contenant la description de l'écran et dans la colonne 7 les valeurs des différentes zones non protégées.

sortie : écran\$.seq.

I\$ et DESC\$ sont écrits dans le fichier écran\$.seq.

REST (restauration du contenu d'un écran).

entrée : écran\$
 sortie : écran\$.séq.

On lit fichier écran\$.séq. L'écran est réinitialisé (même principe au INI excepté que la valeur courante des zones est affichée en lieu et place des tirets).

La description de l'écran est sauvée à nouveau sans les valeurs de la colonne 7 qui est remise à blanc.

OBLI.

entrée : DESC\$.
 sortie : OBLI, vecteur contenant les numéros de zones obligatoires restées vides.

On parcourt toutes les zones. Si elles ont été définies 'obligatoires' (DESC\$ (,8) et si la zone est restée vide (DESC\$ (,7) vide) le numéro de la zone est rangé dans OBLI.

DATE (validation d'une date).

entrée : date\$ date à analyser.
 sortie : coderet.

On découpe la date en année, mois, jour. On vérifie que des bornes sont respectées : 1 mai 19 .

On vérifie le jour par rapport au mois (30 février, 31 avril,...)

Si une erreur est détectée, coderet = 1.

BORNE.

entrée : nozone, numéro de la zone à valider.

sortie : coderet.

On vérifie si la valeur entrée pour cette zone est compatible avec la valeur minimum ou maximum qui ont été définies pour cette zone (DESC\$ (,11) et DESC\$ (,12)).

Si ce n'est pas le cas, coderet = 1.

WRITE (affichage dans une zone).

entrée : Dom\$, nom de la zone dans laquelle une valeur va être affichée.

valeur\$ valeur à afficher.

sortie : coderet.

On vérifie d'abord que l'on va afficher dans une zone non protégée.

Si ce n'est pas le cas coderet = 1.

Grâce au nom de la zone, Dom\$, on recherche par comparaison avec DESC\$ (,2) de quelle zone il s'agit.

Sinon, on se positionne au début de la zone (DESC\$ (,4) et DESC\$ (,5) contiennent les coordonnées de début de la zone) et on affiche la valeur.

READ.

entrée : nozone, numéro de la zone dans laquelle l'utilisateur va écrire.

help.idx, fichier contenant les textes d'aide.

sortie : coderet.

Si la zone considérée est une zone protégée, coderet = 1.

Si l'utilisateur écrit dans cette zone non vide, celle-ci est d'abord réinitialisée.

Tant que l'utilisateur entre des caractères valides (alphanumériques) et que la longueur maximale de la zone n'est pas atteinte, ceux-ci sont affichés dans la zone et caténés à la valeur de la variable correspondant à la zone.

Si le caractère saisi est un point d'interrogation, le contenu de l'écran est sauvé, le texte de help est affiché, puis l'écran est restauré.

Si le caractère saisi est < ret > (l'utilisateur marque ainsi la fin de saisie d'un écran) codret = 3.

Si le caractère n'est pas valide (n'est pas de ceux exposés ci-dessus) et n'est pas un caractère de déplacement dans l'écran (" → " et " ← "), ce caractère n'est pas pris en compte.

Si le caractère saisi est " <ret > " ou " → " ou " ← " ou si la longueur maximale de la zone est atteinte, et si la zone a été définie de type date ou numérique

les différents modules de validation sont appelés.

Si tout est bon, soit on retourne au programme appelant (`ret`) ou on saisit la zone suivante que l'utilisateur a sélectionné simplement en se positionnant dans l'écran à l'aide des touches. " `→` " et " `←` "

4.2.5.3 Primitives de description.

Les primitives de description consistent en un simple jeu de questions-réponses. Nous n'allons donc pas les détailler ici, il suffit d'aller consulter le mode d'emploi des programmes d'initialisation en annexe II.

Le sauvetage de la description d'un écran, écran\$ est une simple écriture du tableau DESC\$ dans le fichier "écran\$.séq."

L'édition d'une image de l'écran à l'imprimante n'est pas très compliquée : au lieu de pratiquer une initialisation à l'écran (INI), les valeurs sont écrites dans un fichier, les espaces libres de l'écran étant remplacés par autant de caractères 'blanc'.

4.3 Lien avec la gestion de la base de données.

Il aurait été idiot de saisir les données à l'écran dans les variables associées aux zones non protégées et puis de transformer le contenu de ces variables dans les variables correspondant aux items définis à l'aide du système de gestion de base de données.

C'est pourquoi nous n'y posons que les zones non protégées destinées à recevoir la valeur d'un item particulier défini via le système de gestion de base de données porte le nom de cet item. Nous évitons ainsi un transfert inutile.

CHAPITRE V : IMPLEMENTATION DE
LA SOLUTION.

5.0 Introduction.

Il n'entre pas dans nos intentions de détailler les algorithmes de toutes les fonctions du logiciel : il s'agit essentiellement d'appel aux primitives de la gestion de la base de données et de la gestion d'écran.

Ces algorithmes se résument pour la plupart à une séquence d'appel à ces différentes primitives. Les énumérer serait fastidieux et sans grand intérêt pour le lecteur. Nous nous contenterons de donner les grandes lignes de ces algorithmes.

5.1 Structure des différents fichiers.

Le fichier "PATIENT" est un fichier indexé relatif :

En effet, on doit pouvoir accéder aux différents enregistrements de ce fichier sous base du numéro d'enregistrement patient qui n'est rien d'autre que le numéro relatif d'enregistrement; mais on doit aussi pouvoir y accéder sous base de différentes clefs (par exemple nous avons composé une clef avec le nom, le sexe et la date de naissance du patient). Rappelons que c'est l'utilisateur qui définit lui-même les clefs d'accès suivant ses besoins (paramétrisation du système).

Le fichier "TUMEUR" est un fichier séquentiel indexé. On ne pouvait choisir le type indexé relatif puisque le numéro d'enregistrement tumeur n'est pas le numéro relatif du record (numéro d'enregistrement tumeur = numéro enregistrement patient catené à l'occurrence de la tumeur).

*Pg 100
utiliser un langage
facile à décrire*

A côté de ces deux registres existent différents fichiers :

- le fichier "help.idx" qui contient les textes d'aide associés aux différentes zones de l'écran. La clef est le nom de la zone.
- les différents fichiers contenant les dictionnaires de code qui sont aussi des séquentiels indexés. Deux clefs ont été définies sur ces fichiers : une sur le code, l'autre sur le libellé.

5.2 Paramétrisation.

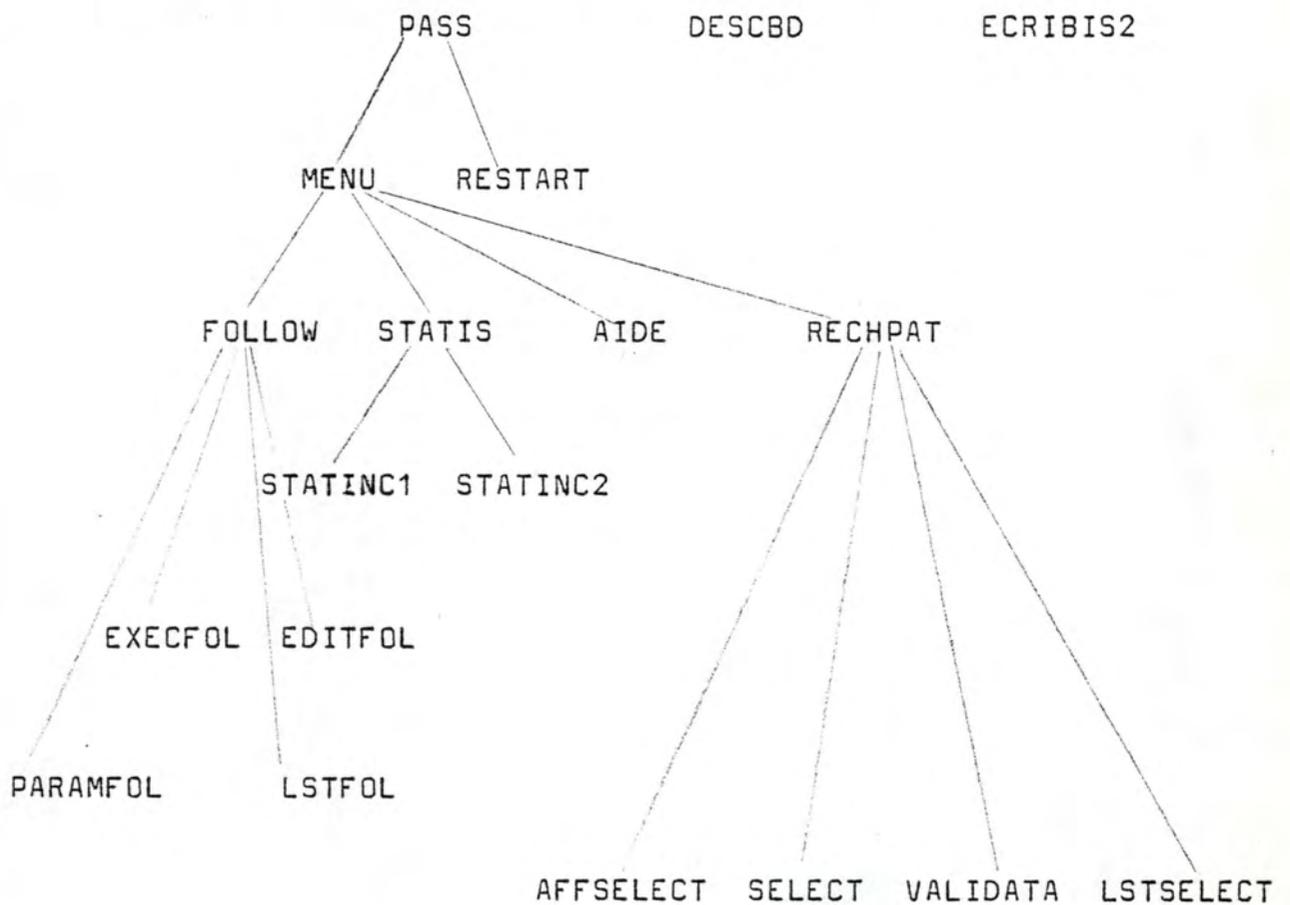
Les programmes sont indépendants du choix des items effectué par le responsable du registre.

La description des fichiers est externe aux programmes. Il suffit que l'utilisateur choisisse parmi la liste des items proposés. Il peut, s'il le désire, en ajouter d'autres mais dans ce cas il lui incombe, si nécessaire, d'ajouter les validations sur les nouvelles données qu'il introduit.

Pour choisir sa structure de données l'utilisateur doit faire appel aux programmes d'initialisation DESCBD et ECRBIS2 (voir annexe II. Mode d'emploi).

5.3 Structure de l'application.

Nous exposons, à l'aide d'un schéma, la structure statique d'appel des différents programmes. Nous exposons ensuite chacun de ces programmes en mentionnant leur effet, les fichiers utilisés et créés.



5.4 Implémentation des différentes fonctions.

5.4.1 PASS (entrée dans le système).

ENTREE :calendr.seq fichier séquentiel contenant la date
du jour et le dernier numéro de patient attribué
pass.seq fichier relatif contenant un mot de passe
de 5 caractères encryptés en 15 caractères
journal.seq fichier de journalisation contenant
les données nécessaires à une éventuelle restauration
de la base de données.

SORTIE :calendr.seq
pass.séq

Le programme teste si le fichier de journalisation est vide.

Si ce n'est pas le cas, il fait appel au programme de restauration RESTART.

On saisit la date du jour et le mot de passe qui sont validés. (*)

Le programme offre la possibilité de modifier le mot de passe.

(*) Si après deux essais le mot de passe fourni par l'utilisateur n'est pas validé, le programme boucle indéfiniment.

5.4.2 RESTART (restauration de la base de données).

Entrée : - journal.seq et les fichiers *.base de données contenant les index à restaurer.

Sortie : journal.seq et *.base de données vide.

On parcourt le fichier de journalisation, on accède aux enregistrements qui étaient en cours de modification quand l'incident qui est à l'origine de cette restauration s'est produit. On réécrit les anciennes valeurs de ces enregistrements. Ces valeurs ont été sauvées dans le fichier de journalisation.

Les différents fichiers d'index sont rétablis : simple copie des fichiers *.base de données qui contiennent les anciennes valeurs d'index.

5.4.3 FOLLOW (follow-up).

entrée : /

sortie : /

Un menu follow-up est affecté et l'aiguillage vers les différentes fonctions est réalisé en fonction du choix de l'utilisateur.

5.4.4 EXECFOL.

Entrée : - calendr.seq
- les fichiers patient et tumeur

Sortie : - follow-dat fichier contenant le numéro des patient pour lesquels on doit éditer une fiche de follow-up. Chacun de ces numéros est suivi des codes des sources auxquelles on doit envoyer ces fiches de follow-up.

Sur base de la date du jour où l'on lit dans le fichier calendr.seq on sélectionne tous les patients vivants pour lesquels on n'a plus de nouvelles depuis au moins un an. Le test s'effectue sur la date de dernier contact. Les numéros de ces patients et les codes des sources auxquelles on doit envoyer les lettres de follow-up sont enregistrés dans le fichier follow-dat.

5.4.5 PARAMFOL (saisie des paramètres d'édition des lettres de follow-up).

Entrée : follow.seq, fichier séquentiel contenant la lettre type de follow-up.

Sortie : Paramfol-seq; fichier contenant les paramètres d'édition du formulaire de follow-up.

Par un système de questions-réponses on saisit et valide les paramètres d'édition de la lettre de follow-up : nombre de lignes avant l'en-tête de la lettre, nombre de lignes entre le texte de la lettre et l'étiquette (adresse de la source).

5.4.6 EDITFOL.

Entrée : paramfol.seq, contenant les paramètres d'édition,
 follow-seq, la lettre type,
 les fichiers patient et tumeur.

Sortie : B : lettre-seq, fichier séquentiel contenant
 les lettres de suivi et prêtes
 à éditer.

On lit le fichier follow-seq. Pour chaque enregistrement de ce fichier, on accède au patient dont le numéro figure dans l'enregistrement. On lit les données nécessaires à l'édition du formulaire de suivi : le nom, la date de naissance du patient, la localisation de sa tumeur.

A l'aide de ces données, de la lettre type et des paramètres d'édition on crée un enregistrement contenant le formulaire de suivi. On y ajoute l'adresse de la source que l'on va chercher dans le dictionnaire adéquat. L'enregistrement est écrit dans le fichier B:lettre.seq.

5.4.7 LSTFOL.

Entrée : B : lettre.seq.

Sortie /

Le fichier B : lettre.seq est vidé à l'imprimante.

5.4.8 STATIS (statistiques).

Entrée : /

Sortie : /

On affiche un menu statistique et en fonction du choix de l'utilisateur on effectue l'aiguillage vers le programme adéquat.

5.4.9 RECHPAT.

Ce programme regroupe la mise à jour et l'interrogation de la base de données. Un menu est d'abord affiché.

Entrée : patient ids et tumeur idx, les fichiers patient et tumeur et les fichiers index correspondant. patient.des et tumeur.des, les fichiers contenant la description des masques de saisie. clendr.seq., fichier contenant la date du jour et le dernier numéro de dossier attribué cancreg num et alp, fichier contenant la description de la base de données.

sortie : les mêmes fichiers éventuellement modifiés.

On affiche d'abord un menu dans lequel on demande à l'utilisateur s'il désire ajouter un cas (1) ou s'il désire interroger, modifier la base de données (2).

-
- (1) - un numéro de dossier est attribué, le fichier calendr.seq est mis à jour,
- le programme de validation est appelé (validata) et on itère sur la saisie et validations des données patient jusqu'à ce que tout soit correct.
 - la grille de saisie des données tumeurs est affichée; le même principe (saisie-validation) est appliqué.

(1) suite.

- quant tout est validé, on enregistre le nouveau cas.

(2)

- après avoir saisi les critères "patient", on saisit les critères tumeurs.
- le programme rechpat fait alors appel au programme de sélection (select) qui va rechercher tous les cas correspondant aux critères saisis.
- rechpat fait ensuite appel au programme affselect qui va afficher un résumé de chacun des cas sélectionnés par le programme select.
- le programme saisit ensuite la sélection de l'utilisateur parmi la liste affichée et suivant le choix de l'utilisateur affiche les cas en détail et permet à ce moment là une éventuelle modification de l'enregistrement (avec validation des données), ou encore liste à l'imprimante des différents cas sélectionnés par l'utilisateur (programme lstselect).
- si un enregistrement a été modifié à ce niveau, il est réécrit.

Dans tous les cas, le programme change ma description de la base de données et des masques de saisie.

Le masque de saisie "patient" est affiché. Les données sont saisies (que ce soient les données d'un nouveau cas ou les critères de recherche.

A partir de ce point, les traitements se différencient.

5.4.10 SELECT (sélection dans la base de données sous base de critères).

entrée : valeurs courantes des différents items de la base de données (ce sont les critères de sélection).

patient.ids et patient.idx, fichier patient et tumeur.

sortie : selct.bd, fichier contenant une liste de couples (numéro patient, numéro tumeur).

Il s'agit des numéros patients et tumeurs qui correspondent aux critères entrés.

Si les critères de recherche sont des valeurs des items constituant les clefs de recherche, la recherche se fait sous base de ces clefs, sinon la recherche est séquentielle dans les fichiers patient et tumeur. Les numéros des cas correspondent aux critères de sélection sont enregistrés dans le fichier selct.bd.

5.4.11 VALIDATA (validation des données).

Entrée : valeurs courantes des différents items de la base de données.

valid.idx, fichier contenant toutes les tables nécessaires aux différents contrôles croisés.

sortie : un message d'erreur et une variable booléenne.

On vérifie la cohérence des données : la chronologie des dates est vérifiée, les contrôles croisés sur le sexe et le site de la tumeur, le site et l'histologie; l'âge du patient et l'histologie, la latéralité et le site de la tumeur sont effectués. On vérifie aussi que les zones obligatoires (cfr. gestion d'écran) ont été remplies.

Différents messages d'erreurs sont générés et un indicateur d'erreur est positionné.

5.4.12 LSTSELECT.

Entrée : les fichiers partient et tumeur sélect.bd,
(cfr. supra 5.4.10).

sortie : /

On imprime un résumé des différents cas dont le numéro se trouve dans sélect.bd:

- lecture en séquence de sélect.bd
- sur base du couple (numéro patient, numéro tumeur) lu accès à l'enregistrement patient et à l'enregistrement tumeur.

- impression des données principales contenues dans ces enregistrements.

5.4.1.3 AFFSELECT.

C'est la même chose que le programme "lstsélect".

Les données sont affichées au lieu d'être imprimées. De plus l'utilisateur effectue une 'sous-sélection' parmi les cas affichés. Les numéros des cas non sélectionnés par l'utilisateur sont effacés du fichier 'select.bd'.

5.4.14 DESCBD et ECRBIS.

(voir chapitre 5, primitives de description de la gestion de la base de données et de la gestion d'écran).

Chapitre VI : Conclusions.

6.0 Introduction.

Le système a été testé pendant environ un mois au registre Vaudois des tumeurs à Lausanne (°).

Il a été initialisé avec les paramètres de ce registre (choix des items et données d'identification du patient). Nous avons expliqué le fonctionnement du système au Docteur Lévi.

Celui-ci a alors utilisé seul le système pendant un mois.

A la fin de cette phase de tests des modifications ont été apportées au système. Ces premières modifications sont la conséquence des remarques du Docteur Lévi.

Nous allons examiner ces remarques auxquelles s'ajouteront nos propres observations. Nous en déduirons les améliorations techniques et fonctionnelles à apporter au système.

(°) Le registre de Lausanne est sous la responsabilité du Docteur Lévi. Le lecteur trouvera une présentation de ce registre dans l'annexe I intitulée "Rapports des visites aux registres de Grenoble et Lausanne".

6.1 Conclusions du test à Lausanne.

Le gros défaut du système est son extrême lenteur en mise à jour et en consultation de la base de données.

Rappelons qu'initialement, le logiciel devait être développé sur un système possédant 128k de mémoire centrale, et qu'en réalité il a été développé sur un matériel ne possédant que 64k de mémoire.

Nous avons dès lors été obligés de travailler en overlay et les différents programmes que le système doit charger successivement sont de gros programmes.

Nous envisageons de développer une version du logiciel sur un 128k. Il existe cependant une autre solution : une nouvelle version de CP/M permet d'adresser plus de 64k. En changeant notre logiciel CP/M contre la nouvelle version, nous pourrions améliorer de façon considérable les performances du système sans apporter de gros changements au logiciel : il ne serait même pas nécessaire de supprimer les chaînages de programmes : avec la nouvelle version de CP/M les programmes restent chargés en mémoire.

Les temps de consultation de la base de données malgré l'amélioration énoncée ci-dessus, avec le système actuel resteront longs. Nous devons, à ce niveau, faire notre "mea culpa" : nous sommes tombés dans le travers classique de l'informaticien : à savoir donner plus à l'utilisateur que ce qu'il avait demandé.

Nous permettons à l'utilisateur d'interroger la base de

données sur n'importe quels critères. Si ces critères sont des valeurs d'attributs qui constituent une des clefs définies, l'accès est instantané.

Dans le cas contraire, les fichiers patients et tumeurs sont balayés séquentiellement et tous les enregistrements répondant aux critères de sélection sont retenus. Il est aisé de s'imaginer le temps que cela peut prendre si on se rappelle que nous travaillons sur un microordinateur.

En pratique, la base de données d'un registre du cancer n'est interrogé que sur quelques critères, dont nous avons d'ailleurs fait des clefs de recherche.

Ces critères sont l'identification du patient (les premières lettres du nom, la date de naissance et le sexe du patient) et le site de la tumeur; ces données sont le plus souvent présentes sur les rapports médicaux; on constate que ces quelques critères permettent une sélection assez fine.

Dès lors, une nouvelle amélioration du système consistera en l'ajout d'un petit module de sélection uniquement sur base des critères énoncés ci-dessus. Cela permettra à l'utilisateur, avant de créer un nouvel enregistrement (nouveau cas), d'interroger la base de données pour savoir rapidement si le patient est déjà connu au nom du registre.

Les autres remarques concernaient le format des listes éditées et les validations.

Selon le Docteur Lévi, un certain nombre de validations doivent être ajoutées : certaines données telles que l'histologie, la cause de la mort (si la date de la mort est connue) sont 'obligatoires' dans le sens où l'utilisateur doit fournir ces données au système pour que le cas puisse être enregistré.

Ces remarques sont valables pour les registres installés dans les pays dits civilisés, mais on ne peut exiger cela dans les pays en voie de développement où le type de données disponibles est dépendant des infrastructures locales.

Dans l'ensemble, les critiques énoncées ci-dessus, mises à part, le logiciel est une bonne réponse au problème posé. Le choix des items s'avère judicieux.

Les personnes qui ont travaillé avec le système ont été fortement impressionnées par la souplesse du système. En très peu de temps on peut se construire un système sur mesure, on peut redessiner à volonté les masques de saisie des données.

Le système est facile à utiliser.

6.2 Nos critiques.

Nous avons commis quelques erreurs dans les choix techniques.

Gestion d'écran.

Les descriptions des différents masques de saisie sont conservées dans des fichiers séquentiels basic qui sont particulièrement vulnérables : si une panne d'alimentation intervient pendant le chargement de ces descriptions on peut perdre le contenu des fichiers.

Gestion de la base de données.(Construction du séquentiel indexé.)

Pour chaque clef 'définie sur un fichier', un fichier d'index est créé. On peut définir au maximum 5 clefs ce qui nous donne cinq fichiers d'index.

Chaque fois que l'on crée un nouvel enregistrement, on doit écrire dans ces cinq fichiers et dans le fichier principal.

Pour des raisons de sécurité, ces fichiers ne peuvent rester ouverts en permanence.

Faute de place mémoire, les index ne peuvent être chargés en mémoire.

Une écriture dans le fichier principal comprend donc 6 ouvertures de fichiers, 6 écritures et 6 fermetures de fichiers. Tout ceci n'est pas particulièrement performant avec un microordinateur.

Examinons les améliorations que nous envisageons.

Gestion d'écran.

Les descriptions des masques de saisies seront rangées dans des fichiers relatifs.

Système de gestion de base de données. *

Nous n'aurions plus qu'un seul fichier, à savoir le fichier principal.

Les premiers enregistrements de ce fichier contiendraient les index.

Actuellement la recherche dans les fichiers d'index est une recherche dichotomique.

Cela implique un certain nombre d'accès dès que nous désirons diminuer. Nous ne savons pas encore très bien comment.

* A l'origine, les programmes de saisie de la description d'un écran ne devaient pas être utilisés par l'utilisateur normal.

Il devait nous servir une seule fois pour décrire une fois pour toutes les grilles de saisie. Nous n'avions dès lors pas introduit toutes les validations nécessaires : validations des données passées au programme, contrôle du non chevauchement de zones ...

On s'est rendu compte par la suite qu'il pouvait être intéressant pour l'utilisateur de définir lui même ses masques de saisie.

Il faudra donc introduire dans les programmes toutes les validations utiles.

6.3 Choix du matériel.

Le matériel choisi est un système multiposte.

Ceci est inutile pour le type de registres auxquels notre système est destiné : ce sont de tout petits registres dont s'occupera une seule personne quelques heures par jour.

Ce matériel ne nous a pas paru très fiable : à plusieurs reprises nous avons perdu quelques fichiers (vive le back up !). *

Ce plus, ce matériel n'est pas très répandu sur le marché mondial. RAIR n'assure pas lui-même le service après-vente. Celui-ci est confié aux firmes qui distribuent le matériel. Cela pourrait poser des problèmes pour l'assistance technique à l'autre bout du monde.

Ce matériel ne serait pas aussi extraordinaire qu'annoncé dicit la rumeur publique.

*et les
stratèg.
(cf p 39)* { Nous conseillons, le Personal Computer d'I.B.M. qui est sûrement la plus classique d'entre toutes les machines de ce type sur le marché.

Sur papier, le Sirius (qui a été revu) et le Zenith 100 semblent 2 machines assez extraordinaires.

* La mémoire disponible par utilisateur est limité à 64k.

Quels autres matériels conseiller ?

Rappelons que notre choix initial s'était porté sur le RAINBOW 100 de Digital.

6.4 Prolongements.

Nous ne reviendrons pas sur les améliorations techniques envisagées.

Des fonctions seront ajoutées au logiciel.

Elles concernent essentiellement les statistiques que nous avons volontairement négligées : ce ne sont pas les fonctions prioritaires.

Nous introduirons les calculs d'incidence, de taux de mortalité, de survie, les calculs des taux standards.

On pourrait aussi ajouter au système une fonction de complémentation des informations : édition automatique de formulaires de demande de complément d'informations pour les cas jugés incomplets.

BIBLIOGRAPHIE.

- Cancer Registration and its techniques
(R. MACLENNAN; C. MUIR; R. STEINITZ; A. WINKLER;
IARC scientific publications n° 21)

- Profil de la morbidité cancéreuse en Algérie
(1966-1975)
Abdenour Yaker (éd. S.N.E.D.)

- Classification internationale des maladies
Revision 75 - Volume 1
(O.M.S. 77)

- Modern Trends in Oncology 1
Part one : RESEARCH PROGRESS
Ed. Ronald W. Rowen. Butterworths.

- Cancer handbook of Epidemiology and Prognosis
Dr J.A.H. Waterhouse
Ed. Churchill Livingstone.

- Classification internationale des maladies
Oncologie 1980 (O.M.S.)

- I.C.D.O. International classification of Diseases
for Oncology (1976 O.M.S.)

- The Cancer Registry of Norway
Survival of Cancer patients
(Landsforiningen mot Kreft - The Norwegian Cancer
Society - OSLO 1975)

- Cancer in South Australia
Incidence, mortality and survival (1977 to 1980)
Incidence and mortality 1980
(South Australian Health Commission)

- Cancer patient survival; report number 5 (1976)
(U.S. department of Health, education and welfare -
Public Health Service National - Institutes of
Health)

- Cancer Medicine
James F. Holland (ND)
Emil Frei III (ND)
(2d edition - Lea and Febiger 1982 Philadelphia)

- C.S.P. procedure manual
Los Angeles County - University of Southern Cali-
fornia - Cancer Surveillance Program
Direction : Thomas M. Mack, M.D.

- Systeme CANCUR
Commission on Cancer of the American College of
Surgeons on the Micronova, Data General mini
computer for the managment of hospital cancer patient
data.

- Data-Base Managment Systems : Powerful Newcomers to
Minicomputer
Michael Gagle and Gary Y. Koehler

- Modèle relationnel - Note de cours. Y.L. Hainaut
F.N.D.P.

- Reliability of Basic Cancer Patient Data.
(Polly Keigl, LINCOLN POLISSAR, WARREN W. LANE and
Vincent Guinée)
(1982 John Wiley and Sons, Ltd)

- Dictionary - Driven Datasystem Introduction
1982 BRENT A. BLUMENSTEIN
Departments of Biometry and Medicine
Emory University Atlanta, Georgia

- Classification internationale type des professions
Bureau international du travail - Genève

- Name search techniques
Bureau of Systems development
(N.Y. State Identification intelligence System
N.Y.S.I.I.S. Document n° 209.

BUMP



0 0 2 3 9 5 7 9 4

*FM B16/1983/25/1

FACULTES UNIVERSITAIRE NOTRE-DAME DE LA PAIX (NAMUR).

INSTITUT D'INFORMATIQUE.

LA GESTION DES REGISTRES DU
CANCER SUR MICROORDINATEUR.

ANNEXES.

Philippe DELFOSSE.

Roselyne GOOSSENS.

Promoteur : J.L. HAINAUT.

Mémoire présenté pour l'obtention
du titre de Licencié et Maître
en Informatique.

Année Académique 1982 - 1983.

TABLE DES MATIERES.pages :Annexe 1 :

Rapports des visites au registre de Grenoble et Lausanne.	1
Visite au registre du cancer de Grenoble (Docteur François Ménégoz).	2
Visite au Centre Vaudois des tumeurs de Lausanne (Docteur Fabio Lévi).	10

Annexe 2 :

CANCREG : Manuel utilisateur.	15
0. Introduction.	17
I. Présentation générale.	17
I.1 Le matériel.	17
I.2 Caractéristiques principales du logiciel.	18
I.3 Entrée dans le système.	20
I.4 Menu principal.	21
I.5 Follow-up (suivi des patients)	22

	<u>pages</u> :
I.5.1 Lancement du follow-up périodique.	22
I.5.2 Modification des paramètres de présentation.	23
I.5.3 Génération des lettres sur floppy.	24
I.5.4 Impression des lettres;	24
I.6 Aide à la codification.	24
I.7 Saisie, recherche et modification.	25
I.7.0 Introduction.	25
I.7.1 Ajout de nouveaux patients.	26
I.7.2 La recherche et la mise à jour.	27
I.8 Statistiques.	28
I.9 Programmes d'initialisation.	29
I.9.0 Introduction.	29
I.9.1 Les fichiers.	29
I.9.2 Initialisation.	35
I.9.2.2 Définition des grilles de saisie des données.	36

pages :

I.10 Découpe de l'application en programmes.	67
I.10.1 Introduction.	67
I.10.2 Schéma de la structure de l'application.	67
I.10.3 Explication de chaque programme.	68
I.10.3.1 Pass.	68
I.10.3.2 Restart.	68
I.10.3.3 Follow.	69
I.10.3.4 Execfol.	69
I.10.3.5 Paramfol.	69
I.10.3.6 Edifol.	70
I.10.3.7 LSTFOL.	70
I.10.3.8 Statis.	71
I.10.3.9 Rechpat.	71
I.10.3.10 Select.	71

	<u>pages</u> :
I.10.3.11 Validata.	72
I.10.3.12 Lstselct.	72
I.10.3.13 Affselect.	73
I.10.3.14 DESCBD et ECRIBIS2.	73
II Guide utilisateur.	74
II.1 Entrée dans le système.	74
II.2 Menu principal.	78
II.3.1 Follow-up.	79
II.3.2 Modification des paramètres de présentation.	80
II.3.3 Génération des lettres de suivi sur floppy.	82

pages :

II.3.4 Impression des lettres de suivi.	83
II.4 Aide à la codification.	84
II.5 Ajout, mise à jour et interrogation de la base des données (RECHPAT).	87
II.6 Statistiques.	95
II.7 Initialisation.	98
II.7.1 Description de la base de données (DESCBD). Choix du contenu des fichiers.	98
II.7.2 Description des grilles de sai- sie.	104

ANNEXE I :

Rapports des visites au registre
de Grenoble et Lausanne.

.

Visite au registre du cancer de Grenoble (Docteur François Ménégoz).

Dans le cadre de nos investigations concernant l'informatisation des registres du cancer sur microordinateur, nous avons été accueillis au centre de Grenoble par le Docteur François Ménégoz.

Nous le remercions de sa précieuse collaboration ainsi que son équipe.

Le Dr Ménégoz nous a d'abord décrit la situation du registre de Grenoble et ses buts.

Il nous a ensuite présenté le système existant (hardware et software).

Nous avons assisté à une démonstration et à partir de là s'est engagée une discussion à savoir ce qui les satisfait dans leur système, les améliorations qu'ils désiraient.

Nous poursuivrons notre exposé par les considérations que cette visite nous a inspirées.

Situation et buts du registre de Grenoble.

Le registre de Grenoble est géré par 4 personnes sous la direction du Dr Ménégos.

Il couvre le département de l'Isère, soit une population de l'ordre de 900 000 habitants.

Le registre a recensé manuellement en 3 ans environ 10.000 cas (y compris des cas antérieurs à 1979, année de création du registre).

Actuellement, il enregistre plus ou moins 2 700 cas par an. Le registre est informatisé depuis 2 mois.

En 1979, lors de la première édition d'un rapport d'incidence, il couvrait 80 % de la population.

Actuellement, ils espèrent approcher un degré de couverture de 90 %.

Leurs principales sources d'information sont la Sécurité Sociale qui leur envoie régulièrement des dossiers et les hôpitaux (essentiellement les laboratoires d'anatomopathologie). Pour cette 2ème source, les médecins vont eux-mêmes chercher l'information.

Quelques mots sur son fonctionnement : ils recouvrent différents documents qu'ils codent suivant des codes standards (ICD-0 par exemple) mais aussi suivant des codes propres (exemple : profession du patient). Toutes ces informations sont notifiées sur un formulaire approprié (voir document en annexe).

Les cas sont ensuite enregistrés sur ordinateur.

Les différents traitements statistiques ne sont pas effectués sur place mais sont faits par l'ordinateur de la Faculté des Sciences de Grenoble (ordinateur HB 68/ Société CII-HB), ordinateur auquel ils sont reliés en permanence).

Leur but essentiel est le calcul de l'incidence en fonction de l'âge, du sexe et de la localisation de la tumeur. Il faut que leur registre fonctionne pendant encore quelques années pour qu'ils puissent envisager le calcul de taux de mortalité.

- - - - -

Description du système existant.

Ordinateur MIC-RAL modèle 8055 (32K)
de REE filiale de CII-HB

Il s'agit d'un ordinateur multipostes-
multitâches (possibilité de 4 écrans-
claviers de 32K chacun).

1 imprimante (pas de possibilité de traitement gra-
phique).

1 disque dur fixe et 1 disque dur amovible de capacité
de 10 Mégabytes chacun.

1 modem pour la liaison avec la Faculté des Sciences.

Le langage utilisé est un BASIC élémentaire orienté ges-
tion (le BAL).

Le système d'exploitation tourne suivant la norme CP/M.

Coût total : 210 000 FF

Du point de vue logiciel, ils disposent de programmes
de saisie et de mises à jour fournis par une firme de
Software.

Coût : 30 000 FF

- - - - -

Commentaires.

a) Mémoire de masse.

On enregistre en moyenne 2 700 nouveaux cas par an.

La longueur d'un enregistrement est de 680 octets, soit un total de 1 836 000 octets.

On n'a pu nous donner aucune estimation sur la taille des dictionnaires de codes enregistrés sur disque.

Ils ont enregistré 3 dictionnaires :

- . le code ICD-0 pour les sites
- . les codes avec les libellés des communes du département de l'Isère
- . une table de contrôle croisé morphologie-topographie contenant les associations impossibles.

Une estimation très grossière et très large nous conduit à moins de 1 Mégabyte.

S'ajoutent à cela les programmes qui prennent peu d'espace-disque; 500K est une estimation très large.

Soit un total estimé de 3,5 Mégabytes.

Supposons une marge de réserve de 100 %, cela nous conduit à 7 Mégabytes.

Nous sommes loin des 20 Mégabytes utilisés.

Le système tel qu'il est là devrait pouvoir enregistrer les cas de 5 ou 6 ans.

b) Logiciel.

Comme précisé plus haut, le logiciel se limite à des programmes de saisie et de mise à jour.

Ces programmes offrent une bonne validation des données notamment le contrôle croisé topographie-morphologie.

On peut bien entendu prévoir des améliorations :

- . La documentation interne de ces programme est réduite au minimum (quasiment inexistante),
- . Aucune aide à la codification n'est offerte (aide qui serait très appréciée selon le Dr Ménégos),
- . En cas de crash, rien n'est prévu : aucune sécurité, aucun programme de reprise, aucune restauration automatique de la base de données.

C'est un minimum que l'on devrait s'attendre à trouver, surtout avec le volume d'informations que l'on a à traiter.

Seule précaution prise une ou deux fois par semaine : un backup est réalisé : copie des disques durs sur l'ordinateur de la Faculté des Sciences.

Mais ce n'est pas automatique.

- . Ils ne disposent d'aucun utilitaire statistique sur leur ordinateur. C'est la grosse machine de la Faculté des Sciences qui effectue ces calculs et qui produit les outputs (package SPSS).
- . Les différents graphiques et histogrammes sont faits à la main.
- . Le follow-up automatique n'est pas encore en place, mais son implémentation est prévue pour le mois de septembre 82.

- - - - -

Considération globale.

Compte tenu du coût de développement du logiciel et compte tenu du hardware disponible, il semble que l'on puisse faire beaucoup mieux.

L'ajout d'un minimum d'utilitaires statistiques ne coûterait pas grand'chose ni en travail, ni en place-mémoire (les programmes prennent peu de place).

Le coût de l'ajout d'une table traçante à leur système serait minime par rapport au coût de l'ensemble de leur système et éviterait un laborieux travail manuel (exemple : traceur graphique HP 7470A : 14 000 FF).

- - - - -

Bibliographie.

Incidence du cancer dans le département de l'Isère,
année 1979
(registre du cancer du département de l'Isère).

Cahier des charges A.R.C.R.E.C.D.I.
Centre Hospitalier Régional de Grenoble
38700 La Tranche.

- - - - -

Visite du Centre Vaudois des tumeurs de Lausanne.

(Docteur Fabio Levi).

Dans le cadre de nos investigations concernant l'informatisation des registres du cancer sur micro-ordinateur, nous avons été accueilli au Centre de Lausanne par le Docteur Levi.

Le Docteur Levi nous a d'abord présenté le Centre et son fonctionnement.

Nous avons suivi l'enregistrement d'un nouveau cas depuis l'arrivée des documents au Centre jusqu'à la saisie des données ce qui nous a permis de nous familiariser avec le logiciel du Centre.

La dernière journée a été consacrée à la partie statistique.

Situation et fonctionnement du registre.

Le registre emploie 3,5 personnes sous la direction du Docteur Levi. Il couvre la population résidente du Canton de Vaud soit 530.000 habitants dont 10 à 12 % d'étrangers. Il enregistre actuellement entre 3.500 et 4.000 cas par an dont 2.200 sont des cas incidents. Au total depuis les débuts du registre en 1972, ils ont enregistré 38.000 cas. Le degré de couverture est actuellement de 90 %. Les trois principales sources d'informations sont :

1. les hôpitaux,
2. les laboratoires,

3. les sources ambulatoires (i.e. les médecins privés).
Le personnel du registre procède lui-même à la collecte des données (rubrique nécrologique des journaux, ...).

Fonctionnement.

L'ensemble des documents reçus concernant un même patient constitue un dossier. A la réception d'un document, quelque soit sa source, on vérifie manuellement si le patient est connu, i.e. possède un dossier (un dossier par tumeur).

Premier cas : le patient est connu.

- on codifie la nouvelle information à même le document reçu,
- si l'information apportée est jugée suffisante, elle est ajoutée à l'enregistrement informatique du cas,
- l'indicatif de complétude du cas est éventuellement modifié suivant l'apport de la nouvelle information.

Deuxième cas : le patient est inconnu.

- on ouvre un dossier, un numéro lui est attribué,
- si l'ensemble des informations contenues dans le nouveau dossier est jugé suffisant par les responsables, le cas est enregistré dans la base de données,
- une valeur est attribuée à l'indicatif de complétude (C si complet, 1 si manque renseignement administratif, 2 si manque renseignement médical).

Dans les deux cas, si l'information est jugée insuffisante le dossier est mis en attente. Un complément d'information peut être demandé (au médecin traitant par exemple) grâce à des formulaires adéquats joints en annexe.

Présentation du système informatique.

Le registre ne possède pas d'ordinateur. Il est relié via un terminal à l'ordinateur IBM du Centre Hospitalier Universitaire du Canton de Vaud (CHUV) et ce pour la saisie, mise à jour des données et interrogation de la base de données. Le registre est en outre relié par un terminal papier au CDC de l'école polytechnique et ce pour la partie statistique.

Logiciel.

Les fonctions offertes sont :

- la saisie des données,
- la mise à jour,
- une interrogation élémentaire de la base de données.

Le follow-up du patient n'est pas prévu. Pour les statistiques, le registre utilise SPSS. Les programmes de saisie et de mise à jour s'accompagnent de validations dans l'ensemble très bien faites.

Remarque : Il n'y a pas d'informaticien en permanence au Centre.

Considération générale et conclusion.

Nous avons pu tester nous-mêmes le système. Nous n'avons pas découvert de grosses anomalies. Le fait d'avoir pu traiter nous mêmes certains cas nous a permis de corriger certaines idées que nous avions.

Nous désirions introduire une aide à la codification. Par exemple, nous pensions enregistrer tout le code ICD-0 (à condition d'avoir suffisamment de place mémoire) et offrir une aide systématique à la codification à l'écran. Il s'avère que c'est beaucoup plus compliqué que nous le pensions, il existe des cas limites que certains mettent dans une classe, d'autres dans une autre; d'autre part les rapports des laboratoires sont parfois ambigus et en tout cas souvent difficiles à interpréter. Cela doit être fait par une personne compétente.

Un autre idée pourrait alors être de créer un fichier dans lequel on mettrait les décisions qui ont été prises dans ces cas ambigus et ces cas limites. Le fichier ferait office de "jurisprudence" pour conduire à une espèce de standardisation. Problème : ce fichier risque de devenir volumineux.

Un effort particulier a été fait du point de vue place mémoire. La taille d'un enregistrement se limite à 126 octets.

L'interrogation de la base de données se fait suivant le nom, l'initiale du prénom et le sexe, ce qui pour les noms courants donne parfois plusieurs écrans. Quand ils peuvent utiliser la date de naissance, il le font. La sélection est alors beaucoup plus fine, mais ils doivent

laisser tomber un autre critère (3 critères maximum pour le moment).

Un patient qui a plusieurs cancers est enregistré plusieurs fois. On a donc des enregistrements avec des numéros différents pour un même patient.

Les informations administratives concernant ce patient sont donc redondantes ce qui n'est pas encore trop grave car ce cas est rare; on ne perd donc pas trop de place mémoire. Mais cela peut devenir ennuyeux le jour où l'on désire modifier un renseignement concernant le patient. On doit le faire dans chaque enregistrement - redondance de traitement.

Remerciements.

Nous remercions vivement le Docteur Levi et toute son équipe pour leur précieuse collaboration et leur grande disponibilité.

Annexe II :

CANCREG : Manuel utilisateur.

Remarque préliminaire.

Certaines parties de l'annexe II sont redondantes par rapport à la partie principale de ce mémoire.

Ces redondances sont inévitables : il fallait exposer brièvement le système aux utilisateurs.

0. Introduction.

Cet exposé comprend deux grandes parties, la première est une présentation générale du logiciel, la seconde est le mode d'emploi détaillé; on y explicitera le contenu des messages apparaissant à l'écran, l'enchaînement des questions-réponses.

I. Présentation générale.

'CANCREG' (Cancer registration) est un logiciel d'aide à la gestion et à l'exploitation des registres du cancer.

Il prend en charge :

- . la saisie des données (avec les validations) à savoir l'ajout, la mise à jour, la suppression des cas individuels,
- . l'interrogation de la base de données tant à l'écran qu'à l'imprimante,
- . le follow-up des patients (édition des lettres de suivi),
- . le calcul et l'édition de résultats statistiques.

I.1 Le matériel.

Le logiciel peut tourner sur n'importe quel microordinateur tournant sous le système d'exploitation CPM (°),

(°) CPM : marque déposée de Digital Research Inc.

possédant 64k bytes de mémoire centrale, 1 disque dur de 5 mégabytes et 1 lecteur de disque souple (essentiellement pour le back-up).

Une imprimante est nécessaire à la réalisation des fonctions telles que édition de listes, de tabulations statistiques.

Nous avons développé le logiciel sur la boîte noire de RAIR, modèle 3/30. Nous présenterons brièvement ce matériel ultérieurement (cfr. appendice II).

Une nouvelle version du logiciel sera disponible dans les prochains mois. Cette version tournera sur un système possédant 128k de mémoire centrale. Ce point ajouté à quelques améliorations du software devrait accroître de façon spectaculaire les performances du système, performances quelque peu décevantes avec un 64k (1).

I.2 Caractéristiques principales du logiciel.

* Le système est destiné à des petits registres. Il permet l'enregistrement de 2.000 cas par an pendant 5 ans. Cette limitation, due essentiellement à la capacité des mémoires de masse, devrait tomber dans les prochains mois; des disques durs de plus grande capacité sont annoncés sur le marché. De plus en plus ces disques sont amovibles ce qui facilitera le back-up de la base de données.

Il existe bien sûr une limite supérieure au-delà de laquelle nous ne pouvons plus garantir les performan-

(1) Initialement le système devait être développé sur un 128k. Ce type de matériel avec disque dur n'était pas disponible sur le marché à l'époque pour 10.000 \$, le budget qui nous était alloué.

ces du système. Cette donnée est encore une inconnue pour le moment.

* Le logiciel est paramétrable.

Nous n'avons trouvé aucune donnée, psychologiquement acceptable, qui permette d'identifier avec certitude un individu dans tous les pays.

Ainsi dans les pays occidentaux, un individu est identifié par son nom, sa date de naissance, son sexe, son adresse...,

dans un pays en voie de développement où ce type de données n'est pas connu avec précision on pourrait travailler par exemple avec le nom, le nom de la tribu, le nom du sorcier du village ...

Nous laissons la possibilité à l'utilisateur de définir à l'initialisation du système (cfr. infra I.9), les données dont il dispose pour identifier un homme.

Le principe est étendu à l'entièreté du système : nous proposons un contenu arbitraire des fichiers; l'utilisateur choisira, à l'initialisation du système, les items qu'il décide de retenir pour son registre.

* Le logiciel contient des programmes de restauration de la base de données. Après des incidents tels que des pannes d'alimentation, la base de données peut être dans un état incohérent (mise à jour en cours inachevée par exemple).

Avant n'importe quelle autre opération, il faut rétablir la base de données dans un état cohérent.

Ces programmes sont automatiques. Des messages indiqueront à l'utilisateur quelles sont les opérations à effectuer ("Recommencer telle opération qui a été interrompue" voir fig.1).

Nous allons maintenant passer en revue les grandes fonctions offertes par ce logiciel et ce sans trop entrer dans les détails : le système est truffé de messages qui guideront l'utilisateur à tout instant. Néanmoins, nous expliciterons ces messages au point II (mode d'emploi détaillé du système).

I.3 Entrée dans le système.

1. S'il y a lieu, dès l'allumage du système, les procédures de restauration automatique de la base de données sont lancées. Lors de cette phase des messages sont affichés indiquant à l'utilisateur qu'il doit par exemple, recommencer la saisie du dernier cas qu'il avait entré.

2. Un mot de passe est demandé : seuls les responsables du registre auront accès à la base de données sous condition de ce mot de passe (les données sur des cas individuels de cancer sont jugées confidentielles).

Le logiciel est fourni avec le mot de passe "FABIO". On peut modifier ce mot de passe. Il est d'ailleurs conseillé de le changer régulièrement.

Dès que l'utilisateur a entré le mot de passe courant le système lui demande s'il désire le modifier. Si oui, le nouveau mot est saisi, avec confirmation.

3. Ensuite la date du jour est demandée. Elle doit être supérieure ou égale à la dernière date entrée. Néanmoins, il peut être utile de pouvoir entrer une date inférieure à la dernière date saisie par le système.

On permettra moyennant confirmation de rentrer une date antérieure.

Cette date sera attribuée automatiquement à chaque cas enregistré ce jour.

(figure 2 et 3)

4. Le menu principal est affiché. (figure 4).

I.4 Menu principal. (MENU)

Les principales fonctions sont affichées, précédées d'un numéro. (fig. 4)

L'utilisateur peut choisir parmi la liste suivante :

- sortir du menu principal (°),
- la mise à jour et interrogation de la base de données,
- le follow-up,
- l'aide à la codification,
- les statistiques.

(°) on se retrouve en mode "commande" sous le système d'exploitation CP/M.

Si l'utilisateur est sorti du menu, il peut y revenir.
L'instruction est "MENU" <ret> .

I.5 Follow-up (suivi des patients). (FOLLOW)

I.5.0 Introduction.

Le but est de sélectionner tous les patients vivants pour lesquels on n'a plus reçu de nouvelles depuis un an. Des formulaires sont édités à destination des sources qui ont fourni ces différents cas. Ils contiennent les données principales concernant le patient et sa tumeur et quelques questions concernant l'état actuel du malade.

L'ensemble des opérations (sélection + édition) peut être assez long. Nous avons scindé le follow en plusieurs phases de façon à permettre d'effectuer la sélection et de lancer plus tard l'impression des formulaires (la nuit par ex.).

Un menu follow-up est affiché. (figure 5).

I.5.1 Lancement du follow-up périodique.

C'est la sélection proprement dite des patients pour lesquels un formulaire devra être édité. Cette phase peut prendre plus d'une minute. Un message est affiché ("Patience") pour avertir l'utilisateur qu'il n'y a rien d'anormal.

La sélection se base sur la différence entre la date du jour (saisie à l'entrée dans le système) et la date du

dernier contact qui fait partie intégrante de l'enregistrement patient.

La sélection s'accompagne de la mise à jour de cette date de dernier contact qui prend comme nouvelle valeur la date du jour.

I.5.2 Modification des paramètres de présentation.

L'utilisateur choisit le format des lettres de suivi (position de l'en-tête, des données sur le patient...).

Rappelons que les programmes sont portables. Or, il peut être utile d'envoyer des caractères de contrôle particulier à l'imprimante (ex. passage en mode qualité lettre); on ne pouvait introduire de tels caractères dans les programmes sous peine de rendre le logiciel dépendant du matériel.

Ces caractères de contrôle sont considérés comme des paramètres de présentation et sont saisis lors de cette phase.

(voir fig. 7, 8 et 9).

Examinons la structure d'une lettre de suivi.

Elle comporte quatre parties :

- une étiquette contenant l'adresse de la source, (a)
- un texte d'introduction, (b)
- une partie identification patient, (c)
- un texte de conclusion. (d)

Les parties (b) et (d) constituent la lettre type qui servira de base à l'édition des formulaires de suivi.

Les parties (a) et (c) sont variables.

L'utilisateur doit marquer l'emplacement de la partie (c) par le caractère " " (clavier Azerty) ou " " (clavier Qwerty), placé en début de ligne.

Ce texte de base est enregistré dans le fichier "FOLLOW.SEQ".

I.5.3 Génération des lettres sur floppy.

Les formulaires à éditer sont générés sur une disquette sur base du sous fichier sélectionné et de la lettre type.

Le fichier contenant ces lettres de suivi porte le nom "B : LETTRE.SEQ".

I.5.4 Impression des lettres.

Le fichier "B:LETTRE.SEQ" est vidé à l'imprimante. Il ne sera détruit qu'au prochain appel à la fonction précédente (cfr. supra I.5.3)

I.6 Aide à la codification. (AIDE)

Cette fonction permet la recherche de valeurs de code sur base de libellés (il n'est pas nécessaire que ces derniers soient complets) et inversement. Il s'agit

essentiellement de la codification internationale des maladies (ICDO) : site et histologie.

S'il le désire, l'utilisateur peut avoir un bref rappel du mode d'emploi de cette fonction.

I.7 Saisie, recherche et modification. (RECHPAT).

I.7.0 Introduction.

Cette fonction permet l'introduction de nouveaux cas, la recherche selon certains critères (interrogation de la base de données), la mise à jour des enregistrements patients et tumeurs.

Après la sélection de la sous-fonction à réaliser, une grille vide est affichée. Si l'utilisateur ne sait pas quoi entrer dans les différents champs, il peut à tout moment taper un point d'interrogation; des explications lui seront fournies.

Le déplacement dans la grille se fait au moyen des touches "→" et "←" en appuyant simultanément sur la touche "shift".

Certains champs, auxquels le système attribue lui-même une valeur, sont interdits. Si l'utilisateur se positionne dans l'un de ces champs, le curseur est renvoyé par le programme à une zone par défaut.

I.7.1 Ajout de nouveaux patients.

On laisse la possibilité à l'utilisateur d'entrer des cas sans vérifier s'ils sont déjà dans les fichiers.

La saisie s'effectuera en deux étapes, d'abord les informations propres au patient, puis celles concernant sa tumeur. Les figures 11 et 12 présentent des exemples de grilles de saisie pour ces deux étapes.

Saisie des données patient :

Un numéro de dossier est attribué automatiquement. D'autres champs sont remplis par le système tels que la date d'enregistrement qui est initialisée à la date du jour.

La complétude, la cohérence des informations sont vérifiées : un minimum de données est demandé pour enregistrer un cas (pour pouvoir le retrouver par la suite). Les données sont validées.

La saisie des données sur la tumeur se fait suivant le même principe.

Exemple de validation : contrôle croisé sur le site de la tumeur et le sexe du patient.

L'utilisateur n'est pas obligé de corriger toutes les "erreurs". Une partie des validations est impérative. On permet de passer outre certaines corrections. Ce type de validation est plus un avertissement qu'un contrôle. Ex. : certains cancers sont improbables

chez les enfants de moins de 10 ans.

Le principe énoncé ci-dessus, à savoir pouvoir passer outre certaines validations, est un moyen d'obliger l'utilisateur à confirmer des valeurs comme celles de l'exemple.

I.7.2 La recherche et la mise à jour.

Cette fonction permet l'interrogation et la mise à jour de la base de données.

La saisie des critères de recherche s'effectue via les mêmes grilles (fig. 11 et 12).

Les critères sont implicitement liés par l'opérateur "et".

On peut donner des renseignements sur le patient, sur la tumeur ou sur les deux.

Des sous-fichiers sont sélectionnés.

L'utilisateur peut visualiser les cas sélectionnés : un récapitulatif de tous les cas correspondant aux critères est affiché.

L'utilisateur peut sélectionner une partie de ces cas pour les mettre à jour ou simplement les consulter. Il peut encore vider la liste sélectionnée à l'imprimante.

C'est aussi une manière pour lui de voir si le cas qu'il désire enregistrer se trouve déjà dans le fichier. Si oui, il le met à jour, sinon il peut demander, après avoir visualisé le récapitulatif des cas répondant aux

critères, l'affichage d'une grille vide et entrer le nouveau cas (les renseignements qu'il avait déjà entré comme valeurs pour les critères ne sont pas perdus, ils sont placés dans les champs appropriés par le système; en résumé l'utilisateur ne doit pas les entrer deux fois).

I.8 Statistiques. (STATIS)

Pour le moment le système ne permet que l'édition de tabulations statistiques de deux types :

- la répartition des tumeurs par date de référence,
- la répartition des tumeurs par classe d'âge.

Le choix se fait via un menu (fig. 18).

Répartition par date de référence. (fig. 19)

Ce programme calcule les occurrences des localisations des tumeurs par année de référence (il s'agit presque toujours de la date du diagnostic).

On demande d'abord d'introduire les classes (années) de l'intervalle pour lequel on désire la répartition.

Le programme calcule les répartitions (ne sont pris en compte que les cas dont le degré de complétude est supérieur ou égal à celui demandé par l'utilisateur; nous expliquerons ultérieurement (cfr. infra I.9.1) ce que nous entendons par là.

Les résultats sont édités à l'imprimante.

Le format est présenté à la figure 30.

Répartition par classes d'âge.

Il existe des classes par défaut. Néanmoins, l'utilisateur peut redéfinir ses classes.

Edition des résultats : voir fig. 31.

I.9 Programmes d'initialisation. (paramétrisation du système).

I.9.0 Introduction.

Ces programmes permettent à l'utilisateur de se construire un système de registre du cancer sur mesure.

Pour faire appel à ces programmes, il faut sortir du menu principal.

Nous proposons un contenu arbitraire de fichier; celui-ci sera détaillé au point I.9.1.

Les programmes d'initialisation permettent à l'utilisateur de retenir une partie de ces items et de donner une signification à d'autres auxquels nous n'avons attribué volontairement aucune sémantique. Nous verrons comment le faire au point I.9.2.

I.9.1 Les fichiers.

Le système travaille sur 2 fichiers, le fichier patient et le fichier tumeur (avec bien sûr des fichiers d'index).

Afin de définir le contenu des fichiers, nous avons visité différents registres en activité.

Ces registres prévoient d'enregistrer plus de données que ce que nous avons retenu mais :

- ou bien ces données ne sont pas utilisées,
- ou bien ces données, souvent trop spécifiques, ne sont pas à leur disposition.

En résumé, nous n'avons retenu que celles qui sont réellement utilisées, en ajoutant quelques items 'libres' sans sémantique propre auxquels chaque utilisateur donnera la signification qu'il désire soit en vue d'études plus particulières, soit qu'il dispose de données supplémentaires (par ex. la religion du patient).

Le fichier Patient.

- no enregistrement patient.(nopatient)

Ce numéro est attribué automatiquement par le système dès l'enregistrement du cas. L'utilisateur ne peut en aucun cas le modifier.

C'est aussi le numéro de dossier du patient. Tous les documents reçus au registre concernant ce malade porteront ce numéro.

- identification = un ensemble d'items destiné à identifier un patient.

Cette zone d'identification comprend deux parties.

La première est fixe (excepté le fait que l'utilisateur doit donner la longueur maximale que peut prendre le

nom; cette donnée variant fort avec les pays).

Elle comprend :

le nom (nompatient), le nom de jeune fille (nomjf),
la date de naissance, le lieu de naissance (datenaiss
et lieunaiss) et le sexe du patient (sexe).

La seconde partie comprend 8 items. Les variables correspondantes ont pour nom ident1 à ident8. Ces items n'ont pas de signification. Ce sont ces variables qui contiendront les données propres à chaque pays et qui permettront l'identification du patient. Leur sémantique et leur longueur seront définies lors de l'appel aux programmes d'initialisation.

Une seule contrainte : on découpe la zone comme on le désire, mais 8 variables max et la somme de leur longueur additionnée aux longueurs des 2 noms doit être inférieure à caractères.

- âge (age).

Cet item est capital; on ne connaît pas toujours la date de naissance avec précision. Dans ce cas, on enregistre une estimation de l'âge à la date d'enregistrement.

Il faut, par conséquent, enregistrer cette date d'enregistrement (recordingdate).

- état-civil (etatcivil).

codification de la situation maritale de l'individu.

- profession (profession).

occupation professionnelle du patient au moment où il contracte la maladie.

Cette donnée est codifiée (classification internationale des professions; on se limite aux grandes subdivisions).

- grand groupe ethnique auquel appartient le patient (ethnie).

- nombre de tumeurs (nbretumeurs).

cet item nous permettra d'accéder aux différentes tumeurs du patient : le numéro d'enregistrement de la tumeur sera la numéro du patient caténé au numéro d'occurrence de la tumeur.

Enfin, on enregistre la date et la cause de la mort (datemort, causemort).

Remarque : entre parenthèses, nous indiquons chaque fois le nom de la variable correspondant à chaque item. Nous verrons ultérieurement pourquoi nous avons été obligés de les mentionner. (cfr. infra.I.9.2)

Le fichier tumeur.

- no enregistrement tumeur (notumeur).

On applique le même principe que pour le numéro patient. (voir aussi l'item 'nombre de tumeurs').

- date de référence (dateref).

il s'agit le plus souvent de la date de diagnostic ou de la date des premiers examens de laboratoire.

- site (site).

codification(ICD0) de la localisation de la tumeur.

- histologie (histo).

codification (ICD0).

- stade de la tumeur (stade).

codification (TNM).

- latéralité de la tumeur (lateralite).

- sources (source 1 à source 5)

on permet d'enregistrer 5 sources pour une même tumeur. On enregistre les codes de ces sources.

- bases du diagnostic.

On précise quelles sont les bases du diagnostic : radiographie, cytologie, autopsie...

(dradio, dcyto, dautop, dbiop, dchirur, dclinic, dbioch, dimuno, dhemato, dautre, dinc pour respectivement : radiographie, cytologie, autopsie, chirurgie, diagnostic purement clinique, biochimie, imunologie, hématologie, autre et inconnu)

Ces variables prennent la valeur 1 et 0 selon que la base du diagnostic soit ce que représente ces variables ou non.

- traitements appliqués.

(tradio, tchimio, tchirurg, thormo, tautre, tneant, tinc pour respectivement radiothérapie, chimiothérapie, chirurgie, hormonothérapie, autre, rien, inconnu). On donne donc à ces variables la valeur 1 ou 0 suivant que le type de traitement a été appliqué ou non.

- degré de completude (degre).

Cet item existe aussi dans le fichier patient. Il exprime le degré de completude des informations. Nous avons décidé d'enregistrer tous les cas, à condition bien sûr que l'on puisse identifier l'individu, même si nous jugeons que ces cas sont incomplets. On aurait pu les enregistrer dans des fichiers intermédiaires en attendant de pouvoir les compléter. Au lieu de cela, nous ajoutons à chaque enregistrement un item qui indiquera si ces cas doivent être pris en compte ou non dans les statistiques. Ce sont les items de base que l'on retrouve dans tous les registres.

Pour permettre des études particulières, nous avons ajouté 6 variables (var1 à var6). Comme ident1 à ident8, ces variables recevront une éventuelle signification lors de l'initialisation du système.

I.9.2 Initialisation.

Remarque catalogue = base de données
 relation = fichier
 domaine = item

La base de données s'appelle "CANCREG". Les fichiers ou encore relations s'appellent "PATIENT" et "TUMEUR".

I.9.2.1 Définition du contenu des fichiers (DESCBD).

Pour simplifier les idées, nous expliquerons directement sur des exemples.

Nous avons dit plus haut que l'utilisateur devait donner lui-même la longueur maximale de l'item 'nompatient'

Détaillons les différentes étapes.

Pour définir le contenu des fichiers, il faut faire appel, après être sorti du menu principal, au programme "DESCBD". Il suffit pour cela de taper le nom du programme suivi de return. S'afficheront alors les noms des bases de données existantes.

On demande à l'utilisateur s'il désire créer une nouvelle base de données (ou catalogue) ou s'il désire modifier une base de données existante.

Dans notre cas il s'agit de modifier "CANCREG", et plus précisément l'item 'NOMPATIENT' qui appartient à la relation 'PATIENT'.

Quand le système demande un nom de relation, on lui donne "PATIENT".

Il demande ensuite un nom de domaine (ou item). On lui

donne "NOMPATIENT".

S'affichent alors les attributs du domaine, à savoir son nom, sa longueur maximale et ce qui est appelé numéro d'identification et qui indique à quelle clé d'accès appartient le domaine. Le système permet la définition de 5 clés d'accès sur chaque fichier.

Une clé serait, par exemple, le nom, le sexe et la date de naissance. Un domaine peut appartenir à plusieurs clés et ne doit pas nécessairement appartenir à une clé.

Dans notre exemple, il s'agit de modifier la longueur. On indique ce que l'on veut faire et on entre la nouvelle valeur.

Remarque importante :

Il est INTERDIT de modifier le nom des variables correspondant aux items définis au point I.9.1. Par contre on peut en ajouter, pour autant que la longueur totale d'un enregistrement dans un fichier ne dépasse pas 255 caractères, ex. : ajout de l'item cause 2.

(cfr. fig. 22)

L'utilisateur peut faire imprimer la description de la base de données.

I.9.2.2 Définition des grilles de saisie des données.
(ECRBIS2)

Il faut adapter les grilles de saisie au contenu des fichiers. Il faut une grille pour saisir un enregistrement patient et une pour un enregistrement tumeur.

Le programme qui vous permet de définir ces grilles s'appelle "ECRBIS2".

Un écran est composé de ZONES de 2 types :
 d'une part les zones PROTEGEES ie les zones dans lesquelles l'utilisateur ne peut pas écrire; ces zones servent d'en-têtes aux zones à remplir par l'utilisateur,
 d'autre part les zones NON PROTEGEES dans lesquelles on peut écrire.

Une zone a un numéro qui est attribué par le système dans l'ordre de création des zones. Elle a un nom. Pour les zones protégées celui-ci n'a aucune importance, on peut même s'abstenir de leur en donner un.

Les zones non protégées ont comme nom, le nom de la variable dont la valeur doit être saisie dans cette zone. (°)

ex. : NOM DU PATIENT : -----

Nous avons deux zones :

la première est protégée, c'est 'NOM DU PATIENT' : elle rappelle à l'utilisateur ce qu'il doit mettre dans la zone qui suit,

la deuxième zone, où sont affichés des tirets pour rappeler à l'utilisateur que c'est là qu'il doit écrire, est destinée à saisir la valeur de la variable 'nom-patient'. Elle portera donc le nom 'nompatient". Cette zone a pour longueur, la longueur maximale que peut avoir une valeur de la variable 'nompatient'.

(°) Ceci est une contrainte impérative.

Les autres attributs de la zone sont saisis. Ce sont la longueur, le libellé (la valeur d'une zone protégée, en fait ce qui va apparaître à l'écran, dans notre exemple 'NOM DU PATIENT'), le type de la zone (numérique, date, alphanumérique)..., le texte d'aide associé à la zone.

A chaque zone peut être associé un texte d'aide qui contiendra toutes les explications nécessaires pour pouvoir remplir la zone. L'utilisateur y aura accès, pendant la saisie des données en tapant un point d'interrogation dans la zone.

A chaque item des fichiers PATIENT et TUMEUR doit correspondre une zone non protégée ayant le même nom que l'item et ayant la même longueur maximale. Chacune de ces zones peut être précédées d'une zone protégée qui servira d'en-tête.

La figure 11 représente une grille de saisie pour un enregistrement patient.

Le programme 'ECRBIS' permet aussi de sortir à l'imprimante une grille vide ie une image de l'écran défini.

De même, on peut éditer un récapitulatif de la description d'un écran.

(cfr. fig. 36)

Prenons un exemple concret :

il s'agit d'initialiser la variable ident1 faisant partie de la zone d'identification du patient. Supposons que l'utilisateur décide qu'elle contiendra le prénom du patient, celui-ci ayant une longueur maximale de 10.

D'abord appel à 'DESCBD' pour donner la longueur. Ensuite appel à ECRIBIS2.

- créer une zone non protégée, de longueur 10 et de nom IDENT1,
- créer une zone protégée juste devant la précédente dans la grille et qui aura comme valeur (ou encore libellé) 'PRENOM DU PATIENT :'.

Dans la deuxième partie, nous réaliserons une initialisation complète (cfr. infra II.7).

Nous créerons ainsi les écrans des figures 11 et 12 et la description de la base de données correspondante.

Nous présentons dans ce paragraphe les séquences questions-réponses, les images principales des écrans, le format des éditions à l'imprimante. Ce sont les figures auxquelles les paragraphes précédents font référence.

Ces schémas seront abondamment commentés dans la deuxième partie de cet exposé intitulée "Guide utilisateur".

RESTAURATION AUTOMATIQUE DE LA BASE DE DONNEES
=====

(message indiquant le Programme a relancer)

fig 1.

ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
ERREUR !!
ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
VOULEZ-VOUS LE MODIFIER (oui, non Par defaut) : ?
DATE DU JOUR (SAAMMJJ) :
DATE INFERIEURE A LA DERNIERE DATE INTRODUITE !!
CONFIRMER (ret = ok, n = not ok) : ?
DATE : ?

fig 2.

DEFINITION D'UN NOUVEAU MOT DE PASSE

ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
ENTREZ ENCORE UNE FOIS CE MEME MOT DE PASSE : ?

fig 3.

MENU GENERAL

0. SORTIR DU MENU GENERAL.
1. FOLLOW-UP.
2. AIDE A LA CODIFICATION.
3. RECHERCHE PATIENT.
4. STATISTIQUE.

VOTRE CHOIX : ?

fig 4.

FOLLOW-UP
=====

0. RETOUR AU MENU PRINCIPAL.
1. LANCER LE FOLLOW-UP PERIODIQUE.
2. MODIFIER LES PARAMETRES DE PRESENTATION.
3. GENERATION DES LETTRES SUR FLOPPY.
4. IMPRESSION DES LETTRES VIA LE FLOPPY.

VOTRE CHOIX : ?

fig 5.

FOLLOW-UP PERIODIQUE
=====

PATIENCE !!

fig 6.

 PARAMETRISATION EN VUE DE L'EDITION DU FOLLOW-UP.
 =====

Les lettres ont un certain format :

- . une étiquette au nom de la source.
- . une introduction.
- . des renseignements concernant le Patient (Nom...)
- . une conclusion.

Nombre de sauts de lignes entre le haut de la Page et le début de l'étiquette : ?

Nombre de blancs à gauche de l'étiquette : ?

Nombre de sauts de lignes entre le bas de l'étiquette et la première ligne du texte d'introduction : ?

fig 7.

 Chaîne de caractères chargée d'initialiser l'imprimante.
 ex Passage en mode qualité lettre.

TAPER RETURN Pour indiquer la fin de l'introduction d'une ligne.

*

Essai à l'(I)mprimante, (E)rreur, (T)ermine.

fig 8.

 RECAPITATULATIF ...

Vous désirez ... ligne(s) blanche(s) avant le début de l'étiquette. Elle sera écartée du bord gauche de la feuille de ... caractères et sera suivie de ... lignes blanche(s).

EST-CE CORRECT (oui, non) ?

fig 9.

 PROGRAMME DE RECHERCHE, LISTAGE, CREATION, MISE A JOUR
 =====

PATIENT ET TUMEUR
 =====

- Ø. RETOUR MENU.
 1. RECHERCHE, AJOUT/MISE A JOUR.
 2. AJOUT PATIENT.

VOTRE CHOIX : ?

fig 10.

 ** REGISTRE VAUDOIS DES TUMEURS **
 =====

NØ RVT : ---- NOM : ----- NOM-J-FILLE : -----

PRENOM1 : -----

PRENOM2 : -----

SEXE : - DATE DE NAISSANCE : ----- DOMICILE : ----

ETAT CIVIL : - ETHNIE : - PROFESSION : ---

DATE D'ENREGISTREMENT : ----- DATE CONTACT : -----

DATE DE DECES : ----- CAUSE DU DECES : 1. ----

2. ----

3. ----

NOMBRE DE TUMEURS : -

fig 11.

```

-----
NO PATIENT : ----- NOM : ----- SEXE : -
DATE DE NAISSANCE : ----- DATE D'ENREGISTREMENT : -----
=====
NO ENREGISTREMENT TUMEUR : --- DATE DE REFERENCE : -----
SITE : ---- HISTOLOGIE : ---- STADE : ----
ETENDUE : --- LATERALITE : -
BASE DU DIAGNOSTIC (degré de completude)
=====
RADIOLOGIE - CYTOLOGIE - AUTOPSIE - BIOPSIE - CHIRURGIE -
CLINIQUE - BIOCHIMIE - IMUNOLOGIE - HEMATO. - AUTRE - INC -

SOURCES : ---- ---- ---- ---- ----
TRAITEMENT(S)
=====
RADIOTHERAPIE - CHIMIO. - CHIRURGIE - HORMONOTHERAPIE -
AUTRE - INCONNU - AUCUN TRAITEMENT -
DEGRE DE COMPLETEUDE -

```

fig 12.

```

-----
NOM *PRENOM * SEXE* NAIS * DATEREF* SITE* DATEDEC* NUM
1. JACOLET JEAN-C M 110230 * 100280 143 11
2. * 100281 144 121181 12
3. JACTIS PAUL M 010250 * 010250 146 41
.
.
.

```

(Ø) ABORT (1) SELECTION (2) AJOUT PATIENT : ?

fig 13.

EDITION DE LISTES

=====

- Ø. RETOUR MENU
 - 1. LISTE NOMINATIVE.
 - 2. LISTE GLOBALE (PATIENT ET TUMEUR)
- fig 14.

* PROGRAMME D'AIDE A LA CODIFICATION *

* ===== *

Voulez-vous le mode d'emploi (Oui, Non Par défaut):? Oui

* Vous pouvez effectuer des recherches de codes ou de *

* libellés dans les dictionnaires sites ci-dessous. *

* Il vous suffit de choisir le dictionnaire et *

* d'introduire une liste de, au plus 10 mots (ou Phra- *

* - ses) clés. Tous les enregistrements de ce dic- *

* tionnaire qui contiennent au moins un mot clé de la *

* liste, peuvent être affichés. *

- Ø. RETOUR MENU.
- 1. SITE (LOCALISATION).
- 2. HISTOLOGIE.
- 3. CAUSE MORT.

fig. 15.

TERMINER LA LISTE AVEC UN MOT CLE NUL ((RETURN))

1 : Pharynx(RET)

2 : uteri(RET)

3 : 174

.

fig 16.

Vous pouvez arrêter la recherche en tapant sur la touche
(DEL)

=====

TYPE : SITE	146
-------------	-----

OROPHARYNX

=====

TYPE : SITE	147
-------------	-----

NASOPHARYNX

=====

TYPE : SITE	148
-------------	-----

HYPOPHARYNX

•
•

=====

TYPE : SITE	174
-------------	-----

FEMALE BREAST

=====

TYPE : SITE	180
-------------	-----

CERVIX UTERI

=====

TYPE : SITE	183
-------------	-----

CORPUS UTERI

•
•

fig 17.

STATISTIQUE

=====

- Ø. RETOUR AU MENU GENERAL.
- 1. REPARTITION PAR DATE DE REFERENCE.
- 2. REPARTITION PAR CLASSES D'AGE.

VOTRE CHOIX : ?

fig 18.

REPARTITION PAR DATE DE REFERENCE

=====

REPARTITION SELON LA DATE DE REFERENCE, LE SEXE ET LA LOCALISATION DE LA TUMEUR.

PREMIERE ANNEE DE REFERENCE A GAUCHE DU TABLEAU 19.. : ?

DERNIERE ANNEE DE REFERENCE A DROITE DU TABLEAU 19.. : ?

Interviendront dans le calcul de répartition uniquement les renseignements pour lesquels le degré de complétude est supérieur ou égal à celui que vous allez introduire.

DEGRE DE COMPLETUDE MINIMUM : ?

IL FAUDRA UN CERTAIN TEMPS !!

fig 19.

REPARTITION PAR CLASSES

=====

REPARTITION SELON LES CLASSES D'AGE ACTUALISE, LE SEXE ET
LA LOCALISATION DE LA TUMEUR.

CLASSE D'AGE PAR DEFAUT.

0 à 4, 5 à 9, 10 à 14, 15 à 19, 20 à 24, 25 à 29, 30 à 34,
35 à 39, 40 à 44, 45 à 49, 50 à 54, 55 à 59, 60 à 64, 65 à
69, 70 à 74, 75 à 79, 80 à 84, 85 à 150.

Voulez-vous effectuer une autre découpe (∅ non, 1 oui) ?
Interviendront dans le calcul de répartition uniquement les
renseignements pour lesquels le degré de completude est
supérieur ou égal à celui que vous allez introduire.

DEGRE DE COMPLETEUDE MINIMUM : ?

IL FAUT UN CERTAIN TEMPS !!

fig 20.

ENTRER SUCCESSIVEMENT LES BORNES SUPERIEURES DES
DIFFERENTES CLASSES. TERMINER EN TAPANT '∅' (ret)

CLASSE : ∅ de ∅ à : ? 10

CLASSE : 1 de 11 à : ? 20

CLASSE : 2 de 21 à : ? ...

.
.

.

et ainsi de suite

fig 21.

 LES FICHIERS DE DESCRIPTIONS SONT :

CANCREG.ALP.

CANCREG.NUM

EXISTE-T-IL DEJA UNE DESCRIPTION ? oui

NOM DU CATALOGUE : ? cancreg

SORTIE IMPRIMANTE DE LA DESCRIPTION CHARGEE ? n

NOM DE LA RELATION : patient

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ? nompatient

ANCIEN PARAMETRES

1. NOM DE DOMAINE : NOMPATIENT

2. NUMERO D'IDENTIFICATION : 1 .

3. LONGUEUR : 30

PARAMETRE A MODIFIER (CR=FIN) : ? 3

NOUVELLE VALEUR : 35

RELATION PATIENT

ANCIEN PARAMETRES

1. NOM DE DOMAINE : NOMPATIENT

2. NUMERO D'IDENTIFICATION : 1

3. LONGUEUR : 35

PARAMETRE A MODIFIER (CR=FIN) : ?

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ? cause2

VOULEZ-VOUS EFFECTIVEMENT AJOUTER UN DOMAINE A CETTE
 RELATION ? oui

NUMERO D'IDENTIFICATION : ? Ø

LONGUEUR DE LA VALEUR DE CE DOMAINE : ? 4

DOMAINE AJOUTE A LA RELATION.

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ?

NOM DE RELATION : ?

SORTIE IMPRIMANTE DE LA DESCRIPTION : ? oui

SAUVETAGE COMMENCE ...

A)

fig 22.

FONCTIONS OFFERTES.

=====

Ø. ARRET

1. SAISIE ET MISE A JOUR DE LA DESCRIPTION D'UN
ECRAN.

2. GENERATION D'UN FICHER CONTENANT LA DESCRIPTION
D'UN ECRAN.

3. GENERATION D'UN FICHER CONTENANT L'IMAGE D'UN
ECRAN.

fig 23.

FONCTIONS OFFERTES.

=====

- Ø. ARRET
- 1. SAISIE ET MISE A JOUR DE LA DESCRIPTION D'UN ECRAN.
- 2. GENERATION D'UN FICHER CONTENANT LA DESCRIPTION D'UN ECRAN.
- 3. GENERATION D'UN FICHER CONTENANT L'IMAGE D'UN ECRAN.

fig. 23

Saisie et maj de la description d'un écran

NOUVEL ECRAN OU MAJ (1-2) ? 1

DONNER LE NOM DE L'ECRAN A GENERER ? PATIENT

fig. 24

 ajout de quelques nouvelles zones protégées à un écran

NOM DE LA ZONE (Ø POUR ARRETER) ?
 SECTION ? 1
 ZONE PROTEGEE (Y-N) ? Y
 COORDONNEE DEBUT DE ZONE ? 20, 1
 LIBELLE ? **registre vaudois des tumeurs**

 NOM DE LA ZONE (Ø POUR ARRETER) ?
 SECTION ? 1
 ZONE PROTEGEE (Y-N) ? Y
 COORDONNEE DEBUT DE ZONE ? 20,2
 LIBELLE ? =====

 NOM DE LA ZONE (Ø POUR ARRETER) ?
 SECTION ? 1
 ZONE PROTEGEE (Y-N) ? Y
 COORDONNEE DEBUT DE ZONE ? 1,3
 LIBELLE ? SEXE :

 NOM DE LA ZONE (Ø POUR ARRETER) ?
 SECTION ? 1
 ZONE PROTEGEE (Y-N) ? Y
 COORDONNEE DEBUT DE ZONE ? 19,3
 LIBELLE ? NOM :

 NOM DE LA ZONE (Ø POUR ARRETER) ?
 SECTION ? 1
 ZONE PROTEGEE (Y-N) ? Y
 COORDONNEE DEBUT DE ZONE ? 37,3
 LIBELLE ? NOM-J-FILLE :

 fig. 25

 ajout de quelques nouvelles zones non protégées
 NOM DE LA ZONE (Ø POUR ARRETER) ? SEXE

SECTION ? 1

ZONE PROTEGEE (Y-N) ? N

COORDONNEE DEBUT DE LA ZONE ? 7,3

LONGUEUR DE LA ZONE ? 1

ZONE OBLIGATOIRE (Y-N) ? Y

TYPE DE LA ZONE (A(alpha)-N(num)-D(date)) ? N

VALEUR MINIMUM ? Ø

VALEUR MAXIMUM ? 2

TEXTE DU HELP (79 CAR MAX PAR LIGNE) (Ø POUR ARRETER)

? Les valeurs reconnues sont les suivantes ...

? Ø. sexe inconnu

? 1. masculin

? 2. féminin

?

NOM DE LA ZONE (Ø POUR ARRETER) ? NOMPATIENT

SECTION ? 1

ZONE PROTEGEE (Y-N) ? N

COORDONNEE DEBUT DE LA ZONE ? 19,3

LONGUEUR DE LA ZONE ? 35

ZONE OBLIGATOIRE (Y-N) ? Y

TYPE DE LA ZONE (A(alpha)-N(num)-D(date)) ? A

VALEUR MINIMUM ?

VALEUR MAXIMUM ?

TEXTE DU HELP (79 CAR MAX PAR LIGNE) (Ø POUR ARRETER)

? introduisez le nom ou une partie seulement (selection)..

? attention aux regles orthographiques à respecter en ce

? qui concerne par exemple :

? les noms composés

? les accents

? les particules

? ...

?

NOM DE LA ZONE (Ø POUR ARRETER) ? NOMJF
SECTION ? 1
ZONE PROTEGEE (Y-N) ? N
COORDONNEE DEBUT DE LA ZONE ? 51,3
LONGUEUR DE LA ZONE ? 15
ZONE OBLIGATOIRE (Y-N) ? N
TYPE DE LA ZONE (A(alpha)-N(num)-D(date)) ? A
VALEUR MINIMUM ?
VALEUR MAXIMUM ?
TEXTE DU HELP (79 CAR MAX PAR LIGNE) (Ø POUR ARRETER)
? même indication que pour le nom
?

fig. 26

 modification des parametres d'une zone
 DONNER LE NUMERO DE LA ZONE A MODIFIER (Ø POUR ARRETER) ? 7

ANCIENNE DESCRIPTION

2. NOM DE LA ZONE	NOMPATIENT
3. CARACTERE PROTEGE DE LA ZONE COORDONNEE DE- BUT DE ZONE	N

4. COLONNE	19
5. LIGNE	3
6. LONGUEUR	35
7. CONTENU	
8. CARACTERE OBLIGATOIRE DE LA ZONE	Y
9. TYPE DE LA ZONE	A
10. SECTION	1
11. PTR ===	6
12. PTR ===	8
13. VALEUR MINIMALE	
14. VALEUR MAXIMALE	

QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? 6
 NOUVELLE VALEUR ? 15

 ANCIENNE DESCRIPTION

2. NOM DE LA ZONE	NOMPATIENT
3. CARACTERE PROTEGE DE LA ZONE COORDONNEES DEBUT DE ZONE	N

4. COLONNE	19
5. LIGNE	3
6. LONGUEUR	15
7. CONTENU	
8. CARACTERE OBLIGATOIRE DE LA ZONE	Y
9. TYPE DE LA ZONE	A
10. SECTION	1
11. PTR <===	6
12. PTR ===>	8
13. VALEUR MINIMALE	
14. VALEUR MAXIMALE	

QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? 15
ERREUR !!

QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? 0
MODIFICATION DU HELP ASSOCIE (Y-N) ? N

fig. 27

Modification, effacement et ajout d'une ligne
d'un texte d'explication d'aide à la saisie

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : ∅. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (∅ POUR ARRETER) ? 90
UN NUMERO COMPRIS ENTRE 100 ET 330 !!
NUMERO DE LA LIGNE A MODIFIER (∅ POUR ARRETER) ? 120
EFFACEMENT OU MAJ (1-2) ? 2
TEXTE DE CETTE LIGNE
? 1. masculin

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : ∅. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (∅ POUR ARRETER) ? 130
AJOUT OU EFFACEMENT (1-2) ? 2
EFFECTUE

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : ∅. SEXE INCONNU
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (∅ POUR ARRETER) ? 120
TEXTE DE CETTE LIGNE
? 1. SEXE MASCULIN

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : ∅. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (∅ POUR ARRETER) ? ∅

fig. 28

DEGRE DE COMPLETEUDE MINIMUM DES INFORMATIONS : 0
 SEXE : INCONNU ou INDEFINI
 DATES ... : LA PREMIERE DATE REPREND LES REPARTITIONS DES ANNEES ANTERIEURES

61.

DATES DE REFERENCE -->	1978	1979	1980	1981	1982	1983	TOTAL
: CODE DIF. DE 140..207	!	-	-	-	-	-	0 F
141: TONGUE	!	-	-	-	-	-	0 F
142: MAJOR SALIVARY GLANDS	!	-	-	-	-	-	0 F
143: GUM	!	-	-	-	-	-	0 F
144: FLOOR OF MOUTH	!	-	-	-	-	-	0 F
145: OTHER AND UNSPECIFIED PARTS OF MO!	!	-	-	-	-	-	0 F
146: OROPHARYNX	!	-	-	-	-	-	0 F
147: NASOPHARYNX	!	-	-	-	-	-	0 F
148: HYPOPHARYNX	!	-	-	-	-	-	0 F
149: PHARYNX AND ILL DEFINED SITES IN!	!	-	-	-	-	-	0 F
150: ESOPHAGUS	!	-	-	-	-	-	0 F
151: STOMACH	!	-	-	-	-	-	0 F
152: SMALL INTESTINE	!	-	-	-	-	-	0 F
153: COLON	!	-	-	-	-	-	0 F
154: RECTUM RECTOSIGMOID JUNCTION ANAL!	!	-	-	-	-	-	0 F
155: LIVER AND INTRAHEPATIC BILE DUCTS!	!	-	-	-	-	-	0 F
156: GALLBLADDER AND EXTRAHEPATIC BILE!	!	-	-	-	-	-	0 F
157: PANCREAS	!	-	-	-	-	-	0 F
158: RETROPERITONEUM AND PERITONEUM	!	-	-	-	-	-	0 F
159: OTHER AND ILL DEFINED SITES WITH!	!	-	-	-	-	-	0 F
160: NASAL CAVITIES ACCESSORY SINUSES !	!	-	-	-	-	-	0 F
161: LARYNX	!	-	-	-	-	-	0 F
162: TRACHEA BRONCHUS AND LUNG	!	-	-	-	-	-	0 F
163: PLEURA	!	-	-	-	-	-	0 F
164: THYRUS HEART AND MEDIASTINUM	!	-	-	-	-	-	0 F
165: OTHER AND ILL DEFINED SITES WITH!	!	-	-	-	-	-	0 F
169: HEMATOPOIETIC AND RETICULOENDOTHE!	!	-	-	-	-	-	0 F
170: BONES JOINTS AND ARTICULAR CARTIL!	!	-	-	-	-	-	0 F
171: CONNECTIVE SUBCUTANEOUS AND OTHER!	!	-	-	-	-	-	0 F
173: SKIN	!	-	-	-	-	-	0 F
174: FEMALE BREAST	!	-	-	-	-	-	0 F
175: MALE BREAST	!	-	-	-	-	-	0 F
179: UTERUS NOS	!	-	-	-	-	-	0 F
180: CERVIX UTERI	!	-	-	-	-	-	0 F
181: PLACENTA	!	-	-	-	-	-	0 F
182: CORPUS UTERI	!	-	-	-	-	-	0 F
183: OVARY FALLOPIAN TURE AND BROAD LI!	!	-	-	-	-	1 F	1 F
184: OTHER AND UNSPECIFIED FEMALE GENI!	!	-	-	-	-	-	0 F
185: PROSTATE GLAND	!	-	-	-	-	-	0 F
186: TESTIS	!	-	-	-	-	-	0 F
187: PENIS AND OTHER MALE GENITAL ORGA!	!	-	-	-	-	-	0 F
188: URINARY BLADDER	!	-	-	-	-	-	0 F
189: KIDNEY AND OTHER URINARY ORGANS !	!	-	-	-	-	-	0 F
190: EYE AND LACRIMAL GLAND	!	-	-	-	-	-	0 F
191: BRAIN	!	-	-	-	-	-	0 F
192: OTHER AND UNSPECIFIED PARTS OF HE!	!	-	-	-	-	-	0 F
193: THYROID GLAND	!	-	-	-	-	-	0 F
194: OTHER ENDOCRINE GLANDS	!	-	-	-	-	-	0 F
195: OTHER ILL DEFINED SITES	!	-	-	-	-	-	0 F
196: LYMPH NODES	!	-	-	-	-	-	0 F
199: UNKNOWN PRIMARY SITE	!	-	-	-	-	-	0 F
----- TOUS LES SITES -----	!	0 F	0 F	0 F	0 F	1 F	0 F

fig.30

REPARTITION SELON LES CLASSES D'AGE ACTUALISE, LE SEXE ET LA LOCALISATION A LA DATE DU : 9830405

DEGRE DE COMPLETUDÉ MINIMUM DES INFORMATIONS : 0
 SEXE : MASCULIN

CLASSES D'AGE --->	AGE																											TOTAL										
	0	4	5	9	10	14	15	19	20	24	25	29	30	34	35	39	40	44	45	49	50	54	55	59	60	64	65		69	70	74	75	79	80	84	85	150	
: CODE DIF. DE 148..287																																						
141: TONGUE																																						
142: MAJOR SALIVARY GLANDS																																						
143: GUM																																						
144: FLOOR OF MOUTH																																						
145: OTHER AND UNSPECIFIED PARTS OF NO																																						
146: DROPHARYNX																																						
147: NASOPHARYNX																																						
148: HYPOPHARYNX																																						
149: PHARYNX AND ILL DEFINED SITES IN																																						
150: ESOPHAGUS																																						
151: STOMACH																																						
152: SMALL INTESTINE																																						
153: COLON																																						
154: RECTUM RECTOSIGMOID JUNCTION ANAL																																						
155: LIVER AND INTRAHEPATIC BILE DUCTS																																						
156: GALLBLADDER AND EXTRAHEPATIC BILE																																						
157: PANCREAS																																						
158: RETROPERITONEUM AND PERITONEUM																																						
159: OTHER AND ILL DEFINED SITES WITHI																																						
160: NASAL CAVITIES ACCESSORY SINUSES																																						
161: LARYNX																																						
162: TRACHEA BRONCHUS AND LONG																																						
163: PLEURA																																						
164: THYMUS HEART AND MEDIASTINUM																																						
165: OTHER AND ILL DEFINED SITES WITHI																																						
169: HEMATOPOIETIC AND RETICULOENDOTHE																																						
170: BONES JOINTS AND ARTICULAR CARTIL																																						
171: CONNECTIVE SUBCUTANEOUS AND OTHER																																						
173: SKIN																																						
174: FEMALE BREAST																																						
175: MALE BREAST																																						
179: UTERUS NOS																																						
180: CERVIX UTERI																																						
181: PLACENTA																																						
182: CORPUS UTERI																																						
183: OVARY FALLOPIAN TUBE AND BROAD LI																																						
184: OTHER AND UNSPECIFIED FEMALE GENI																																						
185: PROSTATE GLAND																																						
186: TESTIS																																						
187: PENIS AND OTHER MALE GENITAL ORGA																																						
188: URINARY BLADDER																																						
189: KIDNEY AND OTHER URINARY ORGANS																																						
190: EYE AND LACRIMAL GLAND																																						
191: BRAIN																																						
192: OTHER AND UNSPECIFIED PARTS OF NE																																						
193: THYROID GLAND																																						
194: OTHER ENDOCRINE GLANDS																																						
195: OTHER ILL DEFINED SITES																																						
196: LYMPH NODES																																						
199: UNKNOWN PRIMARY SITE																																						
----- TOUTS LES SITES -----																																						

Fig 31

1 2 3 4 5
123456789012345678901234567890123456789012345678901234...

```
*****  
* ** REGISTRE VAUDOIS DES TUMEURS ** *  
* ===== *  
* SEXE : - NOM : ----- NOM-J-FILLE : ----- *  
* *  
*****
```

fig. 34

PATIENT

NOMBRE DE ZONES : 8

NØ DE LA ZONE : 1

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 20,1

LONGUEUR DE LA ZONE : 34

ZONE PROTEGEE

LIBELLE : **REGISTRE VAUDOIS DES TUMEURS**

NØ DE LA ZONE : 2

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 20,2

LONGUEUR DE LA ZONE : 34

ZONE PROTEGEE

LIBELLE : =====

NØ DE LA ZONE : 3

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 1,3

LONGUEUR DE LA ZONE : 5

ZONE PROTEGEE

LIBELLE : SEXE

NØ DE LA ZONE : 4

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 12,3

LONGUEUR DE LA ZONE : 5

ZONE PROTEGEE

LIBELLE : NOM :

NØ DE LA ZONE : 5

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 37,3

LONGUEUR DE LA ZONE : 13

ZONE PROTEGEE

LIBELLE : NOM-J-FILLE

NØ DE LA ZONE : 6

SECTION : 1

NOM DE LA ZONE : SEXE

COORDONNEE DEBUT DE ZONE : 7,3

LONGUEUR DE LA ZONE : 1

ZONE NON PROTEGEE

ZONE OBLIGATOIRE

ZONE NUMERIQUE

PTRS	==>	<==
	8	7

HELP ASSOCIE :

? Les valeurs reconnues sont les suivantes ...

? Ø. sexe inconnu

? 1. masculin

? 2. feminin

NØ DE LA ZONE : 7

SECTION : 1

NOM DE LA ZONE : NOMPATIENT

COORDONNEE DEBUT DE ZONE : 19,3

LONGUEUR DE LA ZONE : 15

ZONE NON PROTEGEE
 ZONE OBLIGATOIRE
 ZONE ALPHANUMERIQUE
 PTRS ===> <===
 6 8

HELP ASSOCIE :
 introduisez le nom ou une partie seulement (selection)..
 attention aux règles orthographiques à respecter en ce
 qui concerne par exemple :

- les noms composés
- les accents
- les particules

....

NØ DE LA ZONE : 8
 SECTION : 1
 NOM DE LA ZONE : NOMJF
 COORDONNEE DEBUT DE ZONE : 51,3
 LONGUEUR DE LA ZONE : 15
 ZONE NON PROTEGEE
 ZONE FACULTATIVE
 ZONE ALPHANUMERIQUE
 PTRS ===> <===
 7 6

HELP ASSOCIE :
 même indication que pour le nom

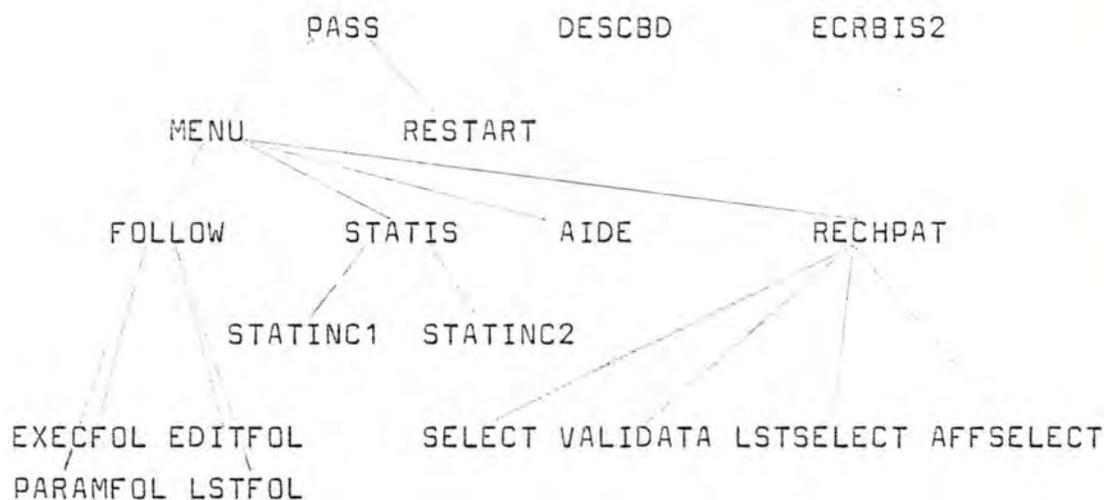
fig. 36

I.10 Découpe de l'application en programmes.

I.10.1 Introduction.

Pour les utilisateurs qui s'intéressent à la manière dont le système est construit, nous présentons brièvement les programmes. Dans un premier temps, nous exposerons au moyen d'un schéma la structure statique d'appel des différents programmes. Ensuite, nous expliquerons succinctement chaque programme en mentionnant son effet, les fichiers utilisés et les fichiers créés.

I.10.2 Schéma de la structure de l'application.



Le programme PASS est lancé automatiquement par CP/M lors de l'allumage du micro-ordinateur. Les programmes DESCBD, DICO et ECRBIS2 sont accessibles uniquement à partir de CP/M, il faut donc pour les faire exécuter, sortir de l'application (CP/M affiche alors les caractères suivant "A>") et taper leur nom.

I.10.3 Explication de chaque programme.

I.10.3.1 PASS.

ENTREE : CALENDR.SEQ : fichier séquentiel-basic contenant la date du jour et le dernier numéro de Patient (numéro de dossier) attribué lors de la création.

PASS.SEQ : fichier relatif-basic contenant un mot de passe de 5 caractères encrypté en 15 caractères.

JOURNALSEQ : fichier relatif-basic de journalisation (destiné à une éventuelle restauration de la base de données).

SORTIE : CALENDR.SEQ modifié avec la date du jour.

PASS.SEQ éventuellement modifié avec un nouveau mot de passe.

EFFET : Si le journal n'est pas vide, PASS fait appel au programme de restauration (RESTART). Il saisit ensuite le mot de passe et la date du jour.

I.10.3.2 RESTART.

ENTREE : JOURNAL.SEQ fichier relatif-basic de journalisation et des fichiers *.B D contenant les valeurs d'index à restaurer.

SORTIE : JOURNAL.SEQ vide.

EFFET : Effectue la restauration de la base de données au moyen du fichier de journalisation. Efface le fichier Journal.seq.

I.10.3.3. FOLLOW.

ENTREE : /

SORTIE : /

EFFET : Affiche le menu du follow-up et lance différents programmes en fonction du choix de l'utilisateur.

I.10.3.4 EXECFOL.

ENTREE : CALENDR.SEQ
 PATIENT.ID- & TUMEUR.ID-
 CANCREG.NUM & ALP description de la base de données.

SORTIE : FOLLOW.DAT Fichier relatif basic.

EFFET : sélection des patients vivants pour lesquels on n'a plus de nouvelles depuis un an et constitue un fichier intermédiaire "FOLLOW.DAT" destiné au programme EDITFOL.

I.10.3.5 PARAMFOL.

ENTREE : FOLLOW.SEQ fichier séquentiel-basic contenant la lettre type du suivi.

SORTIE : PARAMFOL.SEQ.fichier relatif-basic contenant les paramètres de présentation de la lettre de suivi.

EFFET : saisie des paramètres d'édition du formulaire de follow-up.

Création de la lettre type qui est stockée dans "FOLLOW.SEQ".

I.10.3.6 EDIFOL.

ENTREE : PARAMFOL.SEQ fichier relatif-basic contenant les paramètres de présentation de la lettre de suivi.

FOLLOW.SEQ fichier séquentiel-basic contenant le texte de la lettre type.

PATIENT.ID- & TUMEUR.ID-

SOURCE.ID- fichier indexe-relatif contenant la liste des sources d'informations.

CANCREG.NUM à ALP description de la base de données.

SORTIE : B:LETTRE.SEQ. fichier séquentiel-basic contenant les lettres de suivi formatées et prêtes à l'impression.

EFFET : Génération des lettres de suivi sur floppy.

I.10.3.7 LSTFOL.

ENTREE : B:LETTRE.SEQ.

SORTIE : /

EFFET : imprime le contenu du fichier B:LETTRE.SEQ.

I.10.3.8 STATIS.ENTREE : /SORTIE : /EFFET : Affiche le menu statistique et lance différents programmes en fonction du choix de l'utilisateur.I.10.3.9 RECHPAT.ENTREE : PATIENT.ID - & TUMEUR.ID- (index).

PATIENT.DES & TUMEUR.DES description des différents écrans (voir fig 11 et 12).

CANCREG.NUM & ALP description de la base de données.

CALENDR.SEQ fichier séquentiel-basic contenant la date du jour et le dernier numéro de dossier attribué.

SORTIE : PATIENT.ID- & TUMEUR.ID- éventuellement modifiés.

CALENDR.SEQ éventuellement modifiés.

EFFET : Affiche le menu de la fig. 4 (cfr. supra I.10) et coordonne les différentes actions de saisie, ajout, mise à jour et recherche dans le fichier PATIENT et TUMEUR.I.10.3.10 SELECT.ENTREE : valeur des domaines de la relation PATIENT et TUMEUR. (critères de recherche).

PATIENT.ID- & TUMEUR.ID- (fichiers d'index)

CANCREG.NUM & ALP description de la base de données.

SORTIE : SELECT.BD fichier relatif-basic contenant une liste de couples, numéro de patient, numéro de tumeur. Il s'agit des numéros des patients et des tumeurs qui vérifient les critères de recherche.

EFFET : Recherche dans le fichier PATIENT et/ou TUMEUR les patients qui répondent aux critères de recherche et mémorise dans le fichier SELECT.BD les numéros de patient et de tumeur correspondant.

I.10.3.11 VALIDATA.

ENTREE : valeurs des domaines de la relation PATIENT et TUMEUR.

CANCREG.NUM & ALP description de la base de données.

VALID.ID- fichier de validation contrôle-croisé.
Date du jour et nom de la relation.

SORTIE : Un message d'erreur et un indicateur booléen.
Cet indicateur est vrai si l'erreur ne doit pas être obligatoirement corrigée.

EFFET : Ce programme valide les informations présentes. Il vérifie la cohérence et la complétude des informations au moyen du fichier VALID.ID-. Il prépare un message d'erreur et initialise un indicateur en conséquence.

I.10.3.12 LSTSELECT.

ENTREE : PATIENT.ID- & TUMEUR.ID-
CANCREG.NUM & ALP description de la base de données.
SELECT.BD

SORTIE : SELECT.BD

EFFET : Ce programme affiche à l'écran une liste de renseignements concernant les patients et les tumeurs dont les numéros sont enregistrés dans le fichier SELECT.BD. (voir fig. 13). Il enlève de ce fichier les couples de valeur non retenues par l'utilisateur.

I.10.3.13 AFFSELECT.

ENTREE : idem

SORTIE : SELECT.BD non modifié.

EFFET : imprime une liste de renseignements concernant les patients et les tumeurs dont les numéros sont enregistrés dans le fichier SELECT.BD.

I.10.3.14 DESCBD et ECRIBIS2.

Nous ne parlerons pas de DESCBD et de ECRIBIS2. Nous voulons éviter toute tentation à l'utilisateur de fouiller dans ces programme. L'entièreté du logiciel repose sur ces programmes.

II. Guide utilisateur.

Nous allons détailler pour chaque fonction, l'enchaînement des questions du système et des réponses de l'utilisateur. Nous expliquerons le sens des éventuels messages d'erreur.

Rappelons que l'utilisateur peut sortir du menu et appeler directement les fonctions simplement en tapant le nom de la fonction suivi du caractère <ret> (touche RETURN ou ENTER).

```
ex.   A )  STATIS  <ret>
      (*)
```

Le nom des fonctions sera rappelé en caractères gras entre parenthèses.

(*) Quand on est en mode commande, ces deux caractères sont affichés.

II.1 Entrée dans le système.

Un programme (PASS) est lancé automatiquement par le système d'exploitation dès l'allumage du système.

Ce programme effectue les éventuelles restaurations de la base de données, saisit le mot de passe et la date du jour.

Détaillons l'enchaînement des questions-réponses.

Notations.

- Les questions seront numérotées Q1, Q2, ... Chaque question sera suivie d'une explication.
- $\longrightarrow Q_i$ signifie que la séquence normale est interrompue; on reprend à la question Q_i . Cela peut être un branchement en avant ou en arrière (réponse incorrecte par exemple).

Q1 : "Entrez un mot de passe en 5 caractères"

Saisie du mot de passe. Celui-ci n'est pas affiché.

Mot de passe valide $\longrightarrow Q2$

Mot de passe erroné : "ERREUR!" $\longrightarrow Q1$

L'utilisateur a droit à deux essais.

Après ces deux essais, la personne est considérée comme un intrus par le système.

Toutes les opérations sont bloquées à partir du clavier.

Pour pouvoir relancer le système, il faut éteindre et rallumer ce dernier.

Q2 : "Voulez-vous le modifier (oui, non) non par défaut"

Il est conseillé de changer régulièrement le mot de passe. Rappelons que le système traite des données confidentielles.

ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
ERREUR !!
ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
VOULEZ-VOUS LE MODIFIER (oui, non par défaut) : ?
DATE DU JOUR (SAAMMJJ) :
DATE INFERIEURE A LA DERNIERE DATE INTRODUITE!!
CONFIRMER (ret = ok, n = not ok) : ?
DATE : ?

fig 2.

ENTREZ UN MOT DE PASSE EN 5 CARACTERES : ?
ENTREZ ENCORE UNE FOIS CE MEME MOT DE PASSE : ?

fig. 3

Réponse affirmative → Q3

Sinon → Q4

Q3 : "Entrez le nouveau mot de passe"

Le nouveau mot de passe est saisi.

On demande d'entrer une deuxième fois le mot de passe pour s'assurer que l'utilisateur est sûr de l'orthographe de ce mot.

"Entrez encore une fois ce même mot de passe"

Si la deuxième valeur entrée est différente de la première → Q3, sinon → Q4.

Q4 : "Date du jour" (SAAMMJJ)".

La date du jour est demandée sous le format :
siècle, année, mois, jour.

Ex. 18 mai 1898 : 8980518

La date est validée.

Date incorrecte → Q4

Sinon le système vérifie que la date entrée est bien supérieure à la dernière date saisie.

Si ce n'est pas le cas, on demande à l'utilisateur de confirmer qu'il désire entrer cette date

"Date inférieure à la dernière date entrée (...)"

"Confirmer (ret = on, n = not ok)"

Ok menu général

not Ok → Q4

II.2 Menu principal (MENU).

- Le système affiche la figure 4 et prend le choix de l'utilisateur.

-
0. SORTIR DU MENU GENERAL.
 1. FOLLOW-UP.
 2. AIDE A LA CODIFICATION.
 3. RECHERCHE PATIENT.
 4. STATISTIQUE.

VOTRE CHOIX : ?

fig 4.

Q : "Votre choix"

L'utilisateur doit entrer un numéro compris entre 0 et 4 indiquant la fonction a effectuer.

Si la valeur introduite est inférieure à 0 ou supérieure à 4 : —→ Q.

Sinon le programme menu lance la fonction.

II.3 FOLLOW-UP (FOLLOW).

Le menu follow-up est affiché.

-
0. RETOUR AU MENU PRINCIPAL.
 1. LANCER LE FOLLOW-UP PERIODIQUE.
 2. MODIFIER LES PARAMETRES DE PRESENTATION.
 3. GENERATION DES LETTRES SUR FLOPPY.
 4. IMPRESSION DES LETTRES VIA LE FLOPPY.

VOTRE CHOIX : ?

fig 5.

même principe que le menu principal.

1. lancer le follow-up périodique : sélection de tous les individus vivants pour lesquels on n'a plus eu de nouvelles depuis au moins un an.
2. modifier les paramètres de présentation : saisie des paramètres d'édition du formulaire de follow-up. Ces paramètres sont stockés, il n'est donc pas nécessaire de les saisir chaque fois que l'on lance le follow-up.
3. génération des lettres sur floppy : le système génère un fichier contenant toutes les lettres de suivi à imprimer.

4. impression des lettres via le floppy : le fichier créé par la fonction numéro 3 est vidé à l'imprimante.

II.3.1 Follow-up périodique.

Cette phase peut être un peu longue. Un message est affiché pour indiquer que cette attente est parfaitement normale.

II.3.2 Modification des paramètres de présentation.

Les lettres ont un certain format :

- . une étiquette au nom de la source.
- . une introduction.
- . des renseignements concernant le Patient (Nom ...).
- . une conclusion.

Nombre de sauts de lignes entre le haut de la page et le début de l'étiquette : ?

Nombre de blancs à gauche de l'étiquette : ?

Nombre de sauts de lignes entre le bas de l'étiquette et la première ligne du texte d'introduction : ?

fig 7.

Chaîne de caractères chargée d'initialiser l'imprimante.
ex Passage en mode qualité lettre.

TAPER RETURN pour indiquer la fin de l'introduction d'une ligne.

*

Essai à l'(I)mprimante, (E)rreur, (T)ermine.

fig 8.

Vous désirez ... ligne(s) blanche(s) avant le début de l'étiquette. Elle sera écartée du bord gauche de la feuille de ... caractères et sera suivie de ... lignes blanche(s).

EST-CE CORRECT (oui, non) : ?

fig 9.

Commentons la figure 7.

Après un bref rappel du format d'une lettre de suivi, on saisit différents paramètres.

Le nombre de ligne entre le haut de page et l'étiquette ne peut dépasser 20.

Le nombre de blancs à gauche de l'étiquette doit être inférieur ou égal à 90.

Le nombre de lignes entre l'étiquette et le début du texte ne peut excéder 10.

Figure 8.

On saisit les caractères de contrôle à envoyer à l'imprimante pour initialiser celle-ci; par exemple passage en mode qualité lettre.

Vous trouverez des données dans la documentation fournie avec votre imprimante.

Il faut introduire une suite de caractères de contrôle terminée par <return>.

Un caractère introduit ne peut être effacé. Néanmoins, si une erreur s'est glissée dans la série de caractères, il suffit de retaper la bonne suite : après introduction des caractères on demande à l'utilisateur si il y a erreur (E), si c'est correct (T) ou s'il désire faire un essai pour vérifier qu'il s'agit des bons caractères.

S'il y a erreur, une nouvelle suite de caractères est saisie.

Figure 9.

On rappelle à l'utilisateur ce qu'il a demandé et on lui demande s'il est satisfait.

Si ce n'est pas le cas, on recommence Tout !

II.3.3 Génération des lettres de suivi sur floppy.

Q1 : "Veuillez introduire une disquette dans le drive B."
Après avoir mis la disquette vierge et formatée en place, taper (ret).

Q2 : "Désirez-vous imprimer le contenu de la disquette ?
(oui - non)"

Quand tous les formulaires ont été générés, l'utilisateur peut les faire imprimer.

On peut très bien ne pas le faire immédiatement car cela prend un certain temps.

Pour imprimer ultérieurement, il suffira de faire appel à nouveau au follow-up via le menu ou la fonction FOLLOW et de choisir la sous fonction d'impression des lettres de suivi.

II.3.4 Impression lettres de suivi.

Après que l'utilisateur a confirmé qu'effectivement il désire imprimer les formulaires, on demande de positionner l'imprimante en début de page.

Après exécution, on retourne au menu follow-up.

II.4 Aide à la codification. (AIDE).

Rappelons que cette fonction permet à partir de codes de trouver les libellés correspondants et inversement. Il n'est pas nécessaire de donner l'entièreté du libellé, quelques caractères suffisent. Cela concerne la codification de la localisation de la tumeur, de l'histologie et de la cause de la mort.

S'il le désire, l'utilisateur peut avoir à l'écran un rappel du mode d'emploi de cette fonction.

```
*****
*
*          PROGRAMME D'AIDE A LA  CODIFICATION.          *
*          =====*
*****
```

Voulez-vous le mode d'emploi (Oui, Non par défaut) : ? Oui

```
*****
*
*  Vous pouvez effectuer des recherches de codes ou      *
*  de libellés dans les dictionnaires cités ci-dessous.*
*
*  Il vous suffit de choisir le dictionnaire et d'in-   *
*  troduire une liste de, au plus 10 mots (ou phrases) *
*  clés. Tous les enregistrements de ce dictionnaire   *
*  qui contiennent au moins un mot clé de la liste,    *
*  peuvent être affichés.                               *
*
*****
```

- Ø. RETOUR MENU
- 1. SITE (LOCALISATION).
- 2. HISTOLOGIE
- 3. CAUSE MORT.

fig. 15

Q1 : "Voulez-vous le mode d'emploi (oui, non par défaut)".

Q2 : Affichage de la liste numérotée des dictionnaires
 1. Site,
 2. Histologie,
 3. Cause de la mort,
 et question "Votre choix : (0 = retour au menu)"

Q3 : exemple figure 16.

1 : ?

on demande un premier mot clé (code ou morceau de libellé).

Si le mot clé est vide retour au menu,
 sinon on demande un deuxième mot clé

2 : ?

etc.

TERMINER LA LISTE AVEC UN MOT CLE NUL (RETURN)

1. : Pharynx RET

2. : uteri RET

3. : 174

.

.

fig. 16.

On peut ainsi rechercher plusieurs codes et libellés
 en même temps.

Tous les enregistrements du dictionnaire qui contiennent au moins un des mots clés de la liste sont affichés.

Le système affiche un écran à la fois? Pour avoir la suite, il suffit de taper sur la touche <RETURN>. Pour arrêter l'affichage enfoncer la touche DELETE .

```

-----
Vous pouvez arrêter la recherche en tapant sur la
touche DEL
=====
TYPE : SITE      146
OROPHARYNX
=====
TYPE : SITE      147
NASOPHARYNX
=====
TYPE : SITE      148
HYPOPHARYNX
=====
.
.
=====
TYPE : SITE      174
FEMALE BREAST
=====
TYPE : SITE      180
CERVIX UTERI
=====
TYPE : SITE      183
CORPUS UTERI
=====
.
.

```

fig 17.

II.5 Ajout, mise à jour et interrogation de la base de données (RECHPAT).

Une seule fonction permet de réaliser la mise à jour et l'interrogation de la base de données.

Pourquoi ?

La saisie des données pour l'ajout d'un nouveau cas, la modification d'un cas ou l'interrogation de la base de données sont réalisées sur le même canevas.

Q1 :

-
- Ø. RETOUR MENU.
 - 1. RECHERCHE, AJOUT/MISE A JOUR.
 - 2. AJOUT PATIENT.

VOTRE CHOIX : ?

fig 10.

Nous ne détaillerons que la fonction n° 1 : recherche ajout, mise à jour : la seconde fonction, ajout simple de cas est réalisée sur le même principe; des étapes ont simplement été 'court-circuitées.

Recherche, ajout, mise à jour.

Remarque : cette fonction n'est pas redondante avec l'ajout simple de patient. Elle permet à l'utilisateur, en interrogeant la base de données, de voir si le patient est déjà connu du registre et dans ce cas d'effectuer une simple mise à jour.

Q2 : une grille patient vide est affichée.

NØ RVT : ---- NOM : ----- NOM-J-FILLE : -----
PRENOM1 : -----
PRENOM2 : -----
SEXE : - DATE DE NAISSANCE : ----- DOMICILE : -----
ETAT CIVIL : - ETHNIE : - PROFESSION : ----

DATE D'ENREGISTREMENT : ----- DATE CONTACT : -----
DATE DE DECES : ----- CAUSE DU DECES : 1.-----
2.-----
3.-----
NOMBRE DE TUMEURS : -

fig 11.

Elle est destinée à la saisie des données propres au patient (La saisie des données propres à la tumeur se fait à l'aide d'une autre grille).

L'utilisateur initialise différentes zones. Les valeurs de ces zones constituent des critères de recherches. Ceux-ci sont implicitement liés par l'opérateur "et".

On se déplace d'une zone à l'autre en pressant simultanément les touches "→" ou "←" et "shift".

Pour dire que l'on a terminé le remplissage de la grille, il suffit d'enfoncer la touche (return)→Q3.

Q3 : "(o) recherche sur tumeur (1) seulement sur patient".

On demande à l'utilisateur s'il désire effectuer la recherche uniquement sur base des critères-patient (répondre 1) ou s'il désire aussi énoncer des critères-tumeur (répondre 0).

Si réponse = 0 → Q4
 = 1 → Q5

Q4 : Une grille-tumeur vide est affichée.

NO PATIENT : ----- NOM : ----- SEXE : -
 DATE DE NAISSANCE : ----- DATE D'ENREGISTREMENT : ---
 =====
 NO ENREGISTREMENT TUMEUR : --- DATE DE REFERENCE : -----
 SITE : ---- HISTOLOGIE : ---- STADE : ---
 ETENDUE : --- LATERALITE : -
 BASE DU DIAGNOSTIC (degré de completude)
 =====
 RADIOLOGIE - CYTOLOGIE - AUTOPSIE - BIOPSIE - CHIRURGIE -
 CLINIQUE - BIOCHIMIE - IMUNOLOGIE - HEMATO. - AUTRE - INC -

SOURCES : ---- ---- ---- ---- ----

TRAITEMENT(S)
 =====
 RADIOTHERAPIE - CHIMIO. - CHIRURGIE - HORMONOTHERAPIE -
 AUTRE - INCONNU - AUCUN TRAITEMENT -
 DEGRE DE COMPLETEUDE -

fig 12.

Les critères-tumeur sont saisis.

→ Q5.

Remarque : on peut aussi effectuer la recherche uniquement sur base de critères-tumeur. Pour cela, il suffit après l'affichage de la grille-patient de presser directement la touche (ret) et de répondre 0 en Q3.

Q5 : La recherche est lancée. Le message "sélection commencée" apparaît.

Si aucun enregistrement ne répond aux critères le message "NOBODY" est affiché; → Q1. Sinon → Q6.

Q6 : Une liste numérotée des patients et des tumeurs répondant aux critères est affichée.

```

-----
NOM *PRENOM * SEXE * NAIS * DATEREF * SITE * DATEDEC* NUM
1. JACOLET JEAN-C M 110230 *100280      143          11
2.                *100281      144    121181    12
3. JACTIS PAUL      M 010250 *010250      146          41
.
.
.
(∅) ABORT  (1) SELECTION  (2) AJOUT PATIENT : ?

```

fig 13.

→ Q7

Q7 : "(∅) abort (1) sélection (2) ajout patient
(3) page suivante"

- (1) l'utilisateur sélectionne parmi les cas affichés ceux qu'il désire visualiser plus en détails
 —→ Q8
- (2) l'utilisateur constate que le cas dont il s'occupe actuellement n'est pas dans la liste; il demande l'ajout de ce patient
 — Q13
- (3) affichage de la suite de la liste
 —→ Q7
- (4) l'utilisateur arrête sa sélection.
 s'il n'a rien sélectionné —→ Q1
 sinon —→ Q8

Q8 : Sélection par l'utilisateur d'un certain nombre de cas parmi ceux répondant aux critères de recherche.

"CHOIX : "

Il faut donner un numéro de la liste.

Si la réponse est un numéro critère —→ Q8

Si la réponse est (ret) —→ Q7 (pour visualiser les autres pages de la liste).

—→ Q9

Q9 : Le premier cas sélectionné par l'utilisateur est affiché (dans la même grille que pour la saisie des données). Ce sont dans un premier temps les informations propres au patient.

"(0) stop (1) tumeur du patient (2) patient suivant
 (3) maj patient (4) efface (5) ajout patient"

Suivant le choix répondre par le numéro qui précède l'action à effectuer (sans (ret)).

- (0) arrêt des modifications et affichages
→ Q15.
- (1) visualisation des données sur la tumeur de ce patient
→ Q10.
- (2) accéder au patient suivant dans la liste sélectionnée par l'utilisateur
→ Q9
- (3) mise à jour des données-patient. Indiquer la fin des modifications par (ret)
→ Q9
- (4) le cas est effacé après confirmation
→ Q12
- (5) l'utilisateur constate que le cas dont il s'occupe actuellement n'est pas dans les fichiers.
Il décide d'ajouter le patient
→ Q13

Q10 : Les données concernant la tumeur du patient sont affichées dans la grille-tumeur
→ Q11

Q11 : "(Ø) tumeur suivante (1) patient suivant
(2) maj tumeur (3) ajout tumeur
(4) efface"

(Ø) affichage de la tumeur suivante
→ Q10

(1) accès au patient suivant de la sélection de l'utilisateur
→ Q9

(2) les données-tumeurs sont mises à jour
→ Q11

(3) on constate qu'il s'agit d'une nouvelle tumeur pour ce patient

→ Q14

(4) la tumeur du patient est effacée du fichier tumeur après confirmation

→ Q12

Q12 : "Confirm"

L'action demandée nécessite une confirmation de votre part (par exemple : effacement d'un cas).

Pour confirmer presser deux fois (ESCAPE)

Sinon n'importe quel caractère

→ Q11 ou Q9

Q13 : Ajout d'un patient.

Une grille vide est affichée. Certains champs tels que le n° de dossier, la date d'enregistrement sont remplis par le système.

L'utilisateur n'a pas accès à certaines zones. Si il se positionne dans une telle zone, le curseur est envoyé par le système dans une zone par défaut. Les données sont validées.

→ Q14

Q14 : Ajout tumeur.

idem Q13.

"(ø) arrêt ajout (1) ajout patient"

Quand les grilles sont remplies on demande à l'utilisateur s'il veut réellement enregistrer le cas.

Q15 : Quand l'utilisateur a demandé l'arrêt de l'affichage et des modifications, avant de revenir au menu (Q1), on lui demande s'il désire imprimer les cas sélectionnés

(0) abort → Q1

(1) impression → Q16

Q16 :

EDITION DE LISTE
=====

- 1 : RETOUR MENU.
- 2 : LISTE NOMINATIVE.
- 3 : LISTE GLOBALE (PATIENT ET TUMEUR)

fig 14.

(1) liste nominative — Q17

(2) liste globale — Q18

Q17 : édition d'une liste nominative des cas sélectionnés, reprenant le nom, le nom de jeune fille, le sexe et le nombre de tumeurs du patient.

→ Q16

Q18 : édition d'une liste plus complète reprenant des données sur le patient et sa(ses) tumeur(s).

II.6 Statistiques. (STATIS)

Q1 : figure 18

"Votre choix : ?"

L'utilisateur désigne par un chiffre la fonction à exécuter

choix = 1 → Q2

choix = 3 → Q3

Q2 : Répartition par date de référence

figure 19.

On établit la répartition par date de référence.

On demande à l'utilisateur l'intervalle de temps sur lequel on doit calculer cette répartition.

(exemple de 1979 à 1983).

On demande ensuite le degré de complétude (cfr. supra

Q3 : Répartition par classes.

figure 20.

Des classes d'âge par défaut sont données. Si l'utilisateur désire une autre découpe en classes il doit donner successivement les bornes supérieures des différentes classes.

figure 21.

Contrainte : il faut au moins une classe.

On peut donner 17 classes au plus.

Quelque soit votre choix, les calculs prennent un temps certain.

L'utilisateur ne doit pas oublier de brancher et positionner l'imprimante en début de page.

STATISTIQUE

=====

- Ø. RETOUR AU MENU GENERAL.
- 1. REPARTITION PAR DATE DE REFERENCE.
- 2. REPARTITION PAR CLASSE D'AGE.

VOTRE CHOIX : ?

fig. 18

REPARTITION PAR DATE DE REFERENCE.

=====

REPARTITION SELON LA DATE DE REFERENCE, LE SEXE ET LA LOCALISATION DE LA TUMEUR.

PREMIERE ANNEE DE REFERENCE A GAUCHE DU TABLEAU 19.. : ?

DERNIERE ANNEE DE REFERENCE A DROITE DU TABLEAU 19.. : ?

Interviendrons dans le calcul de répartition uniquement les renseignements pour lesquels le degré de complétude est supérieur ou égal à celui que vous allez introduire.

DEGRE DE COMPLETUDE MINIMUM : ?

IL FAUDRA UN CERTAIN TEMPS !!

fig. 19

REPARTITION PAR CLASSES

=====

REPARTITION SELON LES CLASSES D'AGE ACTUALISE, LE SEXE ET
LA LOCALISATION DE LA TUMEUR.

CLASSE D'AGE PAR DEFAUT.

0 à 4, 5 à 9, 10 à 14, 15 à 19, 20 à 24, 25 à 29, 30 à 34,
35 à 39, 40 à 44, 45 à 49, 50 à 54, 55 à 59, 60 à 64, 65 à
69, 70 à 74, 75 à 79, 80 à 84, 85 à 150.

Voulez-vous effectuer une autre découpe (Ø non, 1 oui) ?
Interviendront dans le calcul de répartition uniquement les
renseignements pour lesquels le degré de completude est
supérieur ou égal à celui que vous allez introduire.

DEGRE DE COMPLETEUDE MIMIMUM : ?

IL FAUT UN CERTAIN TEMPS !!

fig. 20

ENTRER SUCCESSIVEMENT LES BORNES SUPERIEURES DES
DIFFERENTES CLASSES. TERMINER EN TAPANT 'Ø' (ret)

CLASSE : Ø de Ø à : ? 10

CLASSE : 1 de 11 à : ? 20

CLASSE : 2 de 21 à : ? ...

.

.

.

et ainsi de suite

fig. 21

II.7 Initialisation.

II.7.0 Nous ne saurions trop conseiller au lecteur de relire attentivement le point I.9.2.

Les paragraphes suivants ne font qu'expliquer comment définir un écran et comment définir le contenu d'une base de données, indépendamment du contexte des registres du cancer.

Néanmoins les exemples utilisés illustreront les contraintes du système et seront tirés de notre application : nous décrirons en effet l'écran patient et le fichier patient correspondant en mettant l'accent sur ces contraintes.

II.7.1 Description de la base de données (DESCBD). Choix du contenu des fichiers.

Q1 : Affichage de la liste des fichiers contenant les descriptions des bases de données existantes et de la question "Existe-t-il déjà une description".
Si vous désirez :

décrire une nouvelle base de données répondre "Y"

→ Q2

modifier une description existante, répondre "W"

→ Q2

Q2 : "Nom du catalogue" (°)

L'utilisateur donne le nom de la base de données dont il désire créer ou modifier la description.

(°) Rappelons que : catalogue = base de données
relation = fichier
domaine = item

Si, l'utilisateur, après avoir répondu "N" à la question Q1 (il désire donc créer une nouvelle base de données) donne un nom de catalogue existant.

Ce dernier sera détruit et les fichiers contenant sa description contiendront la nouvelle description.

Si création → Q3

Si modification → Q4

Q3 : "Nombre de relations maxi : ?"

On demande à l'utilisateur le nombre maximum de relations ou fichiers qui seront créés dans cette base de données.

Un descripteur de la base de données est créé.
Le message "Catalogue ajouté" est affiché :

→ Q4

Q4 : "Nom d'une relation : ?"

L'utilisateur doit fournir le nom du fichier dont il va donner une description (ajout d'une relation au catalogue) ou modifier la description.

Si réponse = <ret> → Q13

Si la réponse est le nom d'une relation existante : (*)

→ Q7

(on considère que l'utilisateur demande une mise à jour)

Si la réponse est le nom d'une nouvelle relation

→ Q5

(on considère que l'utilisateur désire créer une nouvelle relation)

(*) le message "cette relation existe déjà..." est affiché.

Q5 : "Relation inconnue, création d'une nouvelle relation".

"Nombre maxi de domaines pour cette relation".

Idem Q3 : on demande le nombre maximum d'items que l'on décrira pour cette relation.

→ Q6

Q6 : "Type d'accès autorisé (seq, rel, idx, idr) ?"

L'utilisateur donne ici le type du fichier qu'il désire créer.

seq pour séquentiel

rel pour relatif

idx pour indexé-séquentiel

idr pour indexé-relatif

Le descripteur de la relation est créé.

"Le message "relation ajoutée" est affiché.

→ Q7

Q7 : "Nom d'un domaine de cette relation".

L'utilisateur donne le nom de l'item dont il désire modifier les paramètres ou dont il désire créer une description.

Si la réponse est <ret> → Q4 (°)

Si la réponse est le nom d'un item existant → Q11

Sinon → Q8

Q8 : "Nom de domaine inconnu pour cette relation".

On considère que l'utilisateur veut créer le nouvel item.

"Voulez-vous effectivement ajouter un domaine ?"

Y → Q9

N → Q7

(°) En donnant une réponse vide à cette question, on indique la terminaison de l'ajout de domaine ou la terminaison des mises à jour.

Q9 : Saisie des paramètres de l'item.

"Numéro identifiant (suite de chiffre de 0 à 5
séparés par des virgules"

On demande à quelles clés d'accès appartient l'item ou domaine.

→ Q10

Q10: "Longueur maximale de la valeur de ce domaine".

L'utilisateur rentre la longueur de la plus grande valeur que peut prendre cet item.

Le descripteur du domaine est créé.

Le message "Domaine ajouté à la relation" est affiché

→ Q7

Q11: Modification des paramètres de l'item.

Les anciens paramètres sont affichés précédés d'un numéro. "Paramètres à modifier (CR = fin) ?"

L'utilisateur désigne le paramètre qu'il désire modifier → Q12

ret ou 0 = fin de modification des paramètres → Q7

Q12: "Nouvelle valeur".

On saisit la nouvelle valeur du paramètre.

Les nouveaux paramètres sont affichés → Q11

Q13: "Impression de la description ?".

L'utilisateur peut éditer à l'imprimante un résumé complet de la description de la base de données.

Répondre Y ou N.

Q14: "Sauvetage commencé"

Le système sauve la description rentrée par l'utilisateur.

Voici un exemple pour clarifier les esprits : figure 22 et 35.

 LES FICHIERS DE DESCRIPTIONS SONT :

CANCREG.ALP

CANCREG.NUM

EXISTE-T-IL DEJA UNE DESCRIPTION ? oui

NOM DU CATALOGUE : ? cancreg

SORTIE IMPRIMANTE DE LA DESCRIPTION CHARGEE ? n

NOM DE LA RELATION : patient

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ? nompatient

ANCIENS PARAMETRES

1. NOM DE DOMAINE : NOMPATIENT

2. NUMERO D'IDENTIFICATION : 1

3. LONGUEUR : 30

PARAMETRE A MODIFIER CR=FIN : ? 3

NOUVELLE VALEUR : 15

RELATION PATIENT

ANCIENS PARAMETRES

1. NOM DE DOMAINE : NOMPATIENT

2. NUMERO D'IDENTIFICATION : 1

3. LONGUEUR : 15

PARAMETRES A MODIFIER CR=FIN : ?

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ? cause2

VOULEZ-VOUS EFFECTIVEMENT AJOUTER UN DOMAINE A CETTE
 RELATION ? oui

NUMERO D'IDENTIFIANT : ? 0

LONGUEUR DE LA VALEUR DE CE DOMAINE : ? 4

DOMAINE AJOUTE A LA RELATION.

RELATION PATIENT

NOM D'UN DOMAINE DE LA RELATION : ?

 NOM DE RELATION : ?

SORTIE IMPRIMANTE DE LA DESCRIPTION : ? oui

SAUVETAGE COMMENCE ...

A>

fig 22.

 Dans l'exemple ci-dessous nous représenterons le récapitulatif de la description d'une base de données, telle qu'elle a été définie par l'exemple de la fig 22

DESCRIPTION DE LA B.D

=====

Nom du catalogue : cancreg

Nombre de relation : 1

Nom de relation	Nom de domaine	num.id.	long.val.
-----	-----	-----	-----
PATIENT IDR	NOMPATIENT	1	15
	CAUSE2	Ø	4
	.	.	.
	.	.	.

fig 35.

II.7.2 Description des grilles de saisie (ECR BIS2).

Q1 :

 FONCTIONS OFFERTES.

=====

- ∅. ARRET
- 1. SAISIE ET MISE A JOUR DE LA DESCRIPTION D'UN ECRAN.
- 2. GENERATION D'UN FICHER CONTENANT LA DESCRIPTION D'UN ECRAN.
- 3. GENERATION D'UN FICHER CONTENANT L'IMAGE D'UN ECRAN.

fig. 23

"Votre choix ?"

L'utilisateur désigne par un chiffre la fonction à exécuter.

- ∅ l'utilisateur se trouve en mode "commandes"
- 1 → Q2
- 2 → Q24
- 3 → Q25

Q2 : "Nouvel écran ou maj (1-2) ?"

On demande si l'utilisateur désire décrire un nouvel écran (répondre 1) ou modifier une description existante (répondre 2). Le programme interprète toute réponse différente de 1 comme une volonté de mise à jour

1 → Q3

2 → Q15

Q3 : "Donner le nom de l'écran à générer".

On demande de donner un nom à la grille que l'on va décrire.

N.B. Si l'utilisateur indique le nom d'un écran décrit antérieurement, la description de celui-ci sera détruite et remplacée par celle que l'utilisateur va introduire.

Le message "La saisie de la description commence" est affiché.

L'utilisateur va saisir les unes après les autres les zones qui composent l'écran.

→ Q4

Q4 : "Nom de la zone (∅ pour arrêter)".

L'utilisateur introduit le nom d'une zone suivi de < ret > .

Pour marquer la fin de la saisie de la description du masque de saisie l'utilisateur introduit la valeur "∅" < ret > .

∅ → Q14

sinon Q5.

Rappel : chaque zone de l'écran destinée à saisir des données doit porter le nom de la variable destinée à contenir cette donnée.

Saisie des paramètres de la zone.Q5 : "Section ?".

L'écran peut être divisé en "morceaux" appelés sections qui peuvent être affichés seuls (la section \emptyset englobe tout l'écran).

On saisit le numéro de la section à laquelle appartient la zone.

→ Q6.

Q6 : "Zone protégée (Y-N)".

Saisie du caractère protégé de la zone.

(cfr. supra

→ Q7.

Q7 : "Coordonnées début de zone".

L'utilisateur introduit deux nombres séparés par une virgule représentant les coordonnées du début de la zone (la colonne puis la ligne $1 \leq$ numéro de colonne ≤ 79 ; $1 \leq$ numéro de ligne ≤ 22)

si zone protégée → Q8

sinon → Q9.

Q8 : "Libellé".

Une zone protégée est destinée à recevoir les entêtes ou des remarques. On saisit le texte qui doit apparaître à l'écran.

→ Q4.

Q9 : "Longueur de la zone".

L'utilisateur a déclaré une zone non protégée.

On saisit la longueur maximale de cette zone.

→ Q10.

Q10 : "Zone obligatoire (Y-N)".

On demande d'indiquer le caractère obligatoire ou non de la zone.

Une zone déclarée 'obligatoire' ne pourra rester vide lors des saisies de données.

→ Q11.

Q11 : "Type de la zone (A(alpha.)-N(num.)-D(date))?".

"A" < ret> pour indiquer que seuls les caractères alphanumériques peuvent être introduits dans la zone.

"N" < ret> pour interdire les caractères autres que numériques.

"D" < ret> pour indiquer que la zone est destinée à la saisie d'une date. La validité de la date rentrée sera vérifiée.

→ Q12.

Q12 : "Valeur minimum ?"
"Valeur maximum ?".

On saisit les valeurs minimales et maximales que peut prendre la valeur introduite dans la zone.

→ Q13.

Q13 : "Texte du help (79 cara.max. par ligne), (# pour arrêter)".

L'utilisateur introduit ligne par ligne un texte d'explication du contenu attendu de la zone.

N.B. Si un texte d'aide a déjà été défini pour une autre zone portant le même nom que la zone que l'on décrit (par exemple une zone d'un autre écran), cette question (Q13) ne sera pas posée.

→ Q14.

Q14 : Le message "Je travaille pour vous gros veinard" est affiché.

Le programme génère des pointeurs internes qui vont permettre, lors de la saisie des données, à l'utilisateur de se déplacer dans la grille

→ Q1.

Q15 : "Donner le nom de l'écran à modifier".

L'utilisateur introduit le nom de l'écran dont il désire modifier la description.

Q16 : "Donner le numéro de la zone à modifier" (∅ pour arrêter").

Lorsque l'écran a été décrit le programme a attribué un numéro à chaque zone. (°)

L'utilisateur introduit le numéro de la zone dont il désire modifier les paramètres. On indique la fin des modifications par ∅ ret .

- ∅ <ret> → Q1

- il n'existe pas de zone portant le numéro → Q17
sinon → Q18.

Q17 : "Nouvelle zone (Y-N)".

Le numéro saisi en Q16 ne correspond pas à une zone existante. On demande alors à l'utilisateur s'il s'agit d'une nouvelle zone que l'on désire ajouter dans l'écran ou si c'est une erreur.

Y → Q4 --- Q13 puis Q16

N → Q16.

Q18 : L'ancienne description de la zone est affichée précédée du message "Ancienne description".

Il s'agit de la liste numérotée des paramètres de la zone. → Q19

(°) Pour connaître ces numéros, il suffit à l'utilisateur de faire imprimer le fichier contenant un récapitulatif de la description de l'écran (voir Q24).

Q19 : "Quel item désirez-vous modifier (∅ pour arrêter)?".

L'utilisateur indique par un numéro le paramètre à modifier.

Une valeur nulle indiquera la fin des modifications pour la zone considérée.

∅ → Q21

sinon → Q20.

Q20 : "Nouvelle valeur".

L'utilisateur introduit la nouvelle valeur du paramètre (réponses attendues voir Q4 à Q12).

→ Q16.

Q21 : "Modification du help associé (Y-N)".

Si à la zone en cours de modification est associé un texte d'aide, on demande à l'utilisateur s'il désire modifier celui-ci.

Y → Q22

N → Q16.

Q22 : Le texte d'aide est affiché. Chaque ligne est précédé d'un numéro

"Numéro de la ligne à modifier (∅ pour arrêter)"

∅ → Q16

sinon → Q23.

Q23 : "effacement ou maj (1-2)".

L'utilisateur indique s'il désire effacer la ligne désignée en Q22 ou s'il veut simplement modifier son contenu.

N.B. : si le numéro introduit en Q22 n'est pas un numéro de ligne existant, on demande à l'utilisateur s'il désire ajouter une nouvelle ligne

Y on désigne cette ligne → Q22

N Q22

"1" (effacement de la ligne) → Q22

"2" (saisie de la nouvelle valeur de la ligne)
→ Q22.

Q24 : "Nom de l'écran".

L'utilisateur indique le nom de l'écran dont il veut un récapitulatif de la description.

Le programme génère ce récapitulatif et le range dans le fichier qui aura comme nom celui de l'écran suivi de ".lst".

Il suffira de le faire lister à l'aide des commandes du système d'exploitation.

Q25 : idem Q24.

On générera ici un fichier contenant une grille 'vide'.

Le fichier aura pour nom celui de l'écran suivi de '.ima'.

ajout de quelques nouvelles zones protégées à un
écran

NOM DE LA ZONE (Ø POUR ARRETER) ?
SECTION ? 1
ZONE PROTEGEE (Y-N) ? Y
COORDONNEE DEBUT DE ZONE , 20,1
LIBELLE ? ** registre vaudois des tumeurs**

NOM DE LA ZONE (Ø POUR ARRETER) ?
SECTION ? 1
ZONE PROTEGEE (Y-N) ? Y
COORDONNEE DEBUT DE ZONE ? 20,2
LIBELLE ? =====

NOM DE LA ZONE (Ø POUR ARRETER) ?
SECTION ? 1
ZONE PROTEGEE (Y-N) ? Y
COORDONNEE DEBUT DE ZONE ? 1,3
LIBELLE ? SEXE

NOM DE LA ZONE (Ø POUR ARRETER) ?
SECTION ? 1
ZONE PROTEGEE (Y-N) ? Y
COORDONNEE DEBUT DE ZONE , 19,3
LIBELLE , NOM :

NOM DE LA ZONE (Ø POUR ARRETER) ?
SECTION ? 1
ZONE PROTEGEE (Y-N) ? Y
COORDONNEE DEBUT DE ZONE ? 37,3
LIBELLE ? NOM-J-FILLE :

ajout de quelques nouvelles zones non protegees

NOM DE LA ZONE (Ø POUR ARRETER) ? SEXE
SECTION ? 1
ZONE PROTEGEE (Y-N) ? N
COORDONNEE DEBUT DE LA ZONE ? 7,3
LONGUEUR DE LA ZONE ? 1
ZONE OBLIGATOIRE (Y-N) ? Y
TYPE DE LA ZONE (A(alpha) - N(num)-D(date)) ? N
VALEUR MINIMUM ? Ø
VALEUR MAXIMUM ? 2
TEXTE DU HELP (79 CAR MAX PAR LIGNE) (POUR ARRETER)
? Les valeurs reconnues sont les suivantes ...
? Ø. sexe inconnu
? 1. masculin
? 2. féminin
?

NOM DE LA ZONE (Ø POUR ARRETER) ? NOMPATIENT
SECTION ? 1
ZONE PROTEGEE (Y-N) ? N
COORDONNEE DEBUT DE LA ZONE ? 19,3
LONGUEUR DE LA ZONE ? 35
ZONE OBLIGATOIRE (Y-N) ? Y
TYPE DE LA ZONE (A(alpha)-N(num)-D(date)) ? A
VALEUR MINIMUM ?
VALEUR MAXIMUM ?
TEXTE DU HELP (79 CAR MAX PAR LIGNE)(POUR ARRETER)
? introduisez le nom ou une partie seulement (sélection)..
? attention aux règles orthographiques à respecter en ce
? qui concerne par exemple :

? - les noms composés
? - les accents
? - les particules
?
?

NOM DE LA ZONE (~~Ø~~ POUR ARRETER) ? NOMJF
SECTION ? 1
ZONE PROTEGEE (Y-N) ? N
COORDONNEE DEBUT DE LA ZONE ? 51,3
LONGUEUR DE LA ZONE ? 15
ZONE OBLIGATOIRE (Y-N) ? N
TYPE DE LA ZONE (A(alpha)-N(num)-D(date)) ? A
VALEUR MINIMUM ?
VALEUR MAXIMUM ?
TEXTE DU HELP (79 CAR MAX PAR LIGNE)(POUR ARRETER)
? même indication que pour le nom
?

fig. 26.

modification des paramètres d'une zone

DONNER LE NUMERO DE LA ZONE A MODIFIER (Ø POUR ARRETER) ? 7

ANCIENNE DESCRIPTION

2. NOM DE LA ZONE	NOMPATIENT
3. CARACTERE PROTEGE DE LA ZONE	N
COORDONNEES DEBUT DE ZONE	

4. COLONNE	19
5. LIGNE	3
6. LONGUEUR	35
7. CONTENU	
8. CARACTERE OBLIGATOIRE DE LA ZONE	Y
9. TYPE DE LA ZONE	A
10. SECTION	1
11. PTR	=== 6
12. PTR	=== 8
13. VALEUR MINIMALE	
14. VALEUR MAXIMALE	

QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? 6

NOUVELLE VALEUR ? 15

ANCIENNE DESCRIPTION

2. NOM DE LA ZONE	NOMPATIENT
3. CARACTERES PROTEGE DE LA ZONE	N
COORDONNEE DEBUT DE ZONE	

4. COLONNE	19
5. LIGNE	3
6. LONGUEUR	15
7. CONTENU	
8. CARACTERE OBLIGATOIRE DE LA ZONE	Y

9. TYPE DE LA ZONE	A
10. SECTION	1
11. PTR ===	6
12. PTR ===	8
13. VALEUR MINIMALE	
14. VALEUR MAXIMALE	

QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? 15
ERREUR !!
QUEL ITEM DESIREZ-VOUS MODIFIER (Ø POUR ARRETER) ? Ø
MODIFICATION DU HELP ASSOCIE (Y-N) ? N

fig. 27

Modification, effacement et ajout d'une ligne d'un
texte d'explication d'aide à la saisie

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : 0. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (0 POUR ARRETER) ? 90
UN NUMERO COMPRIS ENTRE 100 ET 330 !!
NUMERO DE LA LIGNE A MODIFIER (0 POUR ARRETER) ? 120
EFFACEMENT OU MAJ (1-2) ? 2
TEXTE DE CETTE LIGNE
? 1. masculin

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : 0. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (0 POUR ARRETER) ? 130
AJOUT OU EFFACEMENT (1-2) ? 2
EFFECTUE

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : 0. SEXE INCONNU
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (0 POUR ARRETER) ? 120
TEXTE DE CETTE LIGNE
? 1. SEXE MASCULIN

100 : LES VALEURS RECONNUES SONT LES SUIVANTES ...
110 : 0. SEXE INCONNU
120 : 1. MASCULIN
130 : 2. FEMININ

NUMERO DE LA LIGNE A MODIFIER (0 POUR ARRETER) ? 0

fig. 28

PATIENT. LST

NOMBRE DE ZONE : 8

NØ DE LA ZONE : 1

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 20,1

LONGUEUR DE LA ZONE : 34

ZONE PROTEGEE

LIBELLE : **REGISTRE VAUDOIS DES TUMEURS**

NØ DE LA ZONE : 2

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 20,2

LONGUEUR DE LA ZONE : 34

ZONE PROTEGEE

LIBELLE : =====

NØ DE LA ZONE : 3

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 1,3

LONGUEUR DE LA ZONE : 5

ZONE PROTEGEE

LIBELLE : SEXE

NØ DE LA ZONE : 4

SECTION : 1

NOM DE LA ZONE :

COORDONNEE DEBUT DE ZONE : 12,3

LONGUEUR DE LA ZONE : 5
 ZONE PROTEGEE
 LIBELLE : NOM

NØ DE LA ZONE : 5
 SECTION : 1
 NOM DE LA ZONE :
 COORDONNEE DEBUT DE ZONE : 37,3
 LONGUEUR DE LA ZONE : 13
 ZONE PROTEGEE
 LIBELLE : NOM-J-FILLE

NØ DE LA ZONE : 6
 SECTION : 1
 NOM DE LA ZONE : SEXE
 COORDONNEE DEBUT DE ZONE : 7,3
 LONGUEUR DE LA ZONE : 1
 ZONE NON PROTEGEE
 ZONE OBLIGATOIRE
 ZONE NUMERIQUE
 PTRS === ===
 8 7

HELP ASSOCIE :

? Les valeurs reconnues sont les suivantes ...

? Ø. sexe inconnu
 ? 1. masculin
 ? 2. féminin

NØ DE LA SONE : 7
 SECTION : 1
 NOM DE LA ZONE : NOMPATIENT
 COORDONNEE DEBUT DE ZONE : 19,3
 LONGUEUR DE LA ZONE : 15

ZONE NON PROTEGEE
 ZONE OBLIGATOIRE
 ZONE ALPHANUMERIQUE

PTRS	===	===
	6	8

HELP ASSOCIE :

introduisez le nom ou une partie seulement (sélection)..
 attention aux règles orthographiques a respecter en ce
 qui concerne par exemple :

- les noms composés
- les accents
- les particules

....

.

NØ DE LA ZONE : 8

SECTION : 1

NOM DE LA ZONE : NOMJF

COORDONNEE DEBUT DE ZONE : 51,3

LONGUEUR DE LA ZONE : 15

ZONE NON PROTEGEE

ZONE FACULTATIVE

ZONE ALPHANUMERIQUE

PTRS	===	===
	7	6

HELP ASSOCIE :

même indication que pour le nom

fig. 36.

Appendices.

I. Taper 'oui' suivi de (ret) signifie qu'après avoir tapé 'oui' il faut enfoncer la touche (RETURN).

- Dans l'énoncé de certaines questions, on expose parfois les réponses possibles.

une réponse du genre (oui ou non (ret), oui par défaut) signifie que si l'on désire répondre oui à la question, il n'est pas nécessaire de taper 'oui'. Il suffit de presser la touche (return). La réponse ne sera considérée comme négative que si on répond effectivement NON suivi de (return) à la question posée.

- ESC = escape
- RET = return
- DEL = delete.

Spécifications de la Boîte Noire 3/30.

Micro-Processeur

Micro-processeur	8085A
Horloge temps réel	2 x fréquence de ligne
Interruptions	4 niveaux
DMA (accès direct mémoire)	4 canaux mode continu
Gestion de la mémoire	Topographie de l'adresse des pages 16 x 8 bits
Option processeur arithmétique	Opérations en virgule fixe et flottante 16 et 32 bits

Mémoire RAM (Mémoire à accès sélectif)

Capacité de base de la mémoire	64 ko
Capacité maximale de la mémoire	512 ko
Incréments d'extension	64 ko
Caractéristiques de la mémoire	Régénération transparente, sans états d'attente

Points de connexion entrée/sortie en série

Type d'interface	RS-232C/CCITT V24 compatible
Vitesse d'exploitation	de 50 à 19.200 bauds
Modes de communication	synchrones/asynchrones
Nombre maximal d'interfaces	16

Disque fixe intégral

Type d'entraînement	Micro-Winchester 5,25"
Mode d'enregistrement	MFM
Capacité	6.380 ko brut
Vitesse de transfert	5.000.000 bits/sec.
Temps d'accès	3ms piste à piste

Floppy disque

Type d'entraînement	Disquette 5,25" à double tête
Mode d'enregistrement	MFM (double densité)
Format d'enregistrement	Implantation des secteurs fixée par programme
Capacité	500 ko brut
Vitesse de transfert	250.000 bits/sec.
Temps d'accès	5ms piste à piste

Option de bus parallèle

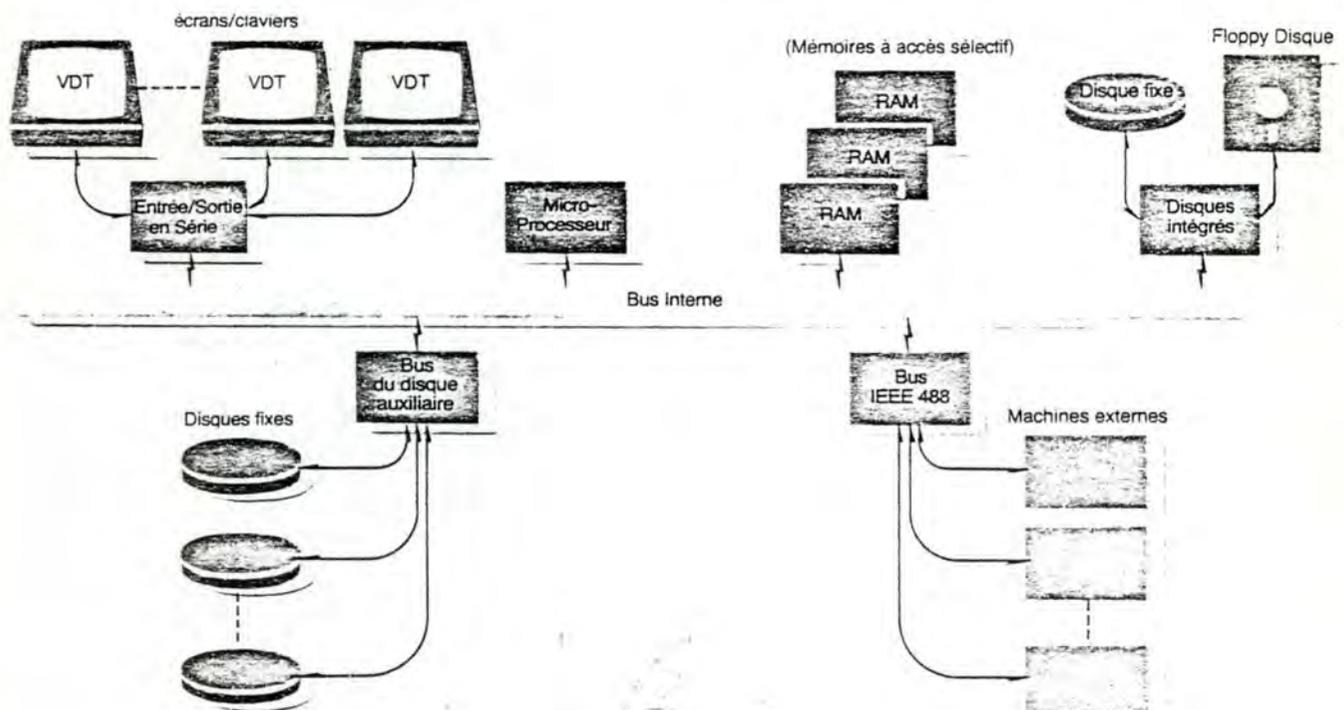
Type d'interface	IEEE 488 standard
Vitesse d'exploitation	800 ko/sec.
Modes d'exploitation	contrôleur/émetteur/écouteur
Longueur maximale de bus	20 mètres

Option de disque auxiliaire

Type d'entraînement	8" Winchester
Capacité d'entraînement	11.500 ko brut
Nombre maximal d'unités	8

Général

Alimentation à découpage	115/230 Vac ± 10%, 50/60 Hz 3,15 A max
Dimensions: Largeur	508 mm
Profondeur	406 mm
Hauteur	152 mm
Ambiance d'utilisation	de 10° à 40° C Humidité relative: de 20 à 80 %
Poids	15,9 kg



III CP/M.

Il n'est pas dans nos intentions d'expliquer toutes les commandes du système d'exploitation CP/M.

Pour cela le lecteur se réfèrera aux manuels que l'on a du lui fournir en même temps que son matériel.

Nous rappelons brièvement trois commandes utiles dans le contexte de notre application.

TYPE filename :
affiche à l'écran le contenu du fichier 'filename'
ex. : type patient.lst provoque l'affichage du
contenu du fichier 'patient.lst'

LIST filename :
imprime le contenu du fichier 'filename' sur
listing.
Ne pas oublier de positionner l'imprimante en
début de page

DIR
affiche la liste des fichiers.

où filename = racine , extension
racine = maximum 8 caractères alphanumériques
extension = maximum 3 caractères alphanumériques.