

# SOLAR

FMS

Systeme de gestion de fichier

LOGICIEL

LOGICIEL

LOGICIEL

LOGICIEL

LOGICIEL

**F M S**

**MANUEL D'UTILISATION**

- △ en haut de page indique le changement complet de la page par rapport à l'IE précédent**
- ! en marge indique la partie modifiée par rapport à l'IE précédent**

## AVERTISSEMENT

Le lecteur trouvera dans cette notice un ensemble de conseils concernant l'utilisation globale du produit FMS.

- Mise en oeuvre pratique. C'est à dire comment intégrer FMS dans un système d'exploitation, pour les différentes versions :

- . FMS-E
- . FMS-G

- Un ensemble d'informations décrivant la structure et le fonctionnement interne de FMS-E, dans sa version de Juillet 1978.

- Quelques conseils permettant d'optimiser l'accès aux fichiers réalisé par FMS et FUP.

Le lecteur se reportera au Manuel de Référence afin d'y trouver :

- Une introduction technique et générale au produit FMS.

- Une description précise des requêtes du système de fichiers et de leur interface de programmation.

<b>SOMMAIRE</b>	<b>Pages</b>
<b>1 - GENERATION</b>	<b>1-1</b>
<b>1.1 - PRESENTATION</b>	<b>1-1</b>
<b>1.2 - DESCRIPTION DU PRODUIT FMS</b>	<b>1-2</b>
<b>1.2.1 - Composition des bibliothèques</b>	<b>1-2</b>
a) Bibliothèque Performance : FMSE - : S	1-2
b) Bibliothèque Méthodes d'accès : FMSEMA - : S	1-3
c) Bibliothèque Extension Noyau : FMSENO - : S	1-3
d) Bibliothèque FMSG - : S	1-3
<b>1.2.2 - Règle générale de construction</b>	<b>1-3</b>
<b>1..2.3 - Les possibilités de choix : FMS-E et FMS-G</b>	<b>1-4</b>
a) Le choix des méthodes d'accès	1-4
b) Le choix des options de performance	1-4
c) Pour FMS-G	1-5
d) Rappel	1-5
<b>1.2.4 - Exemples de choix élémentaires FMS-E</b>	<b>1-6</b>
<b>1.3 - GENERATION DE FMS</b>	<b>1-9</b>
<b>1.3.1 - Principe général</b>	<b>1-9</b>
a) Des modules programmes	1-9
b) Un module table : TABFMS - : S	1-9
<b>1.3.2 - Schéma de génération</b>	<b>1-10</b>
<b>1.3.3 - Exemple de production de TABFMS</b>	<b>1-11</b>
a) Phase de macro-définition	1-11
b) Phase de macro-génération	1-11
c) Génération de la TDFU	1-11
d) Choix à la carte du système de fichiers	1-11
e) Phase d'assemblage de TABFMS	1-12
<b>1.3.4 - Règles de fabrication de TABFMS</b>	<b>1-12</b>
a) Phase de macro-définition	1-12
b) Phase de génération de TABFMS	1-12
c) Génération de la TDFU	1-12
d) Choix du Système de fichier	1-14
<b>1.3.5 - Génération de la TDFU</b>	<b>1-15</b>
a) TDFU	1-15
b) DPU : Descripteur de PU	1-16
c) DFU : Descripteur de FU	1-16
d) Fabrication de la TDFU	1-17

<b>2 - STRUCTURE INTERNE FMS-E</b>	<b>2 - 1</b>
<b>2.1. - PRESENTATION</b>	<b>2 - 2</b>
2.1.1. - Schéma d'un Système d'Exploitation	2 - 2
2.1.2. - FMS-E et les éléments Link-édités	2 - 3
2.1.3. - Description générale des modules	2 - 4
a) Le Superviseur	2 - 4
b) Les modules de FMS-E	2 - 4
<b>2.2. - DESCRIPTION GENERALE</b>	<b>2 - 6</b>
2.2.1. - Environnement d'une Méthode d'Accès dans FMS-E	2 - 6
2.2.2. - Description générale des interfaces	2 - 7
a) Interface Superviseur / FMS	2 - 7
b) Interface FMS / OPEN-CLOSE	2 - 7
c) Interface OPEN-CLOSE / Noyau (MOC)	2 - 7
d) Interface Noyau (MDK) / IOCS	2 - 7
e) Interface FMS / Méthode d'Accès	2 - 7
f) Interface OPEN-CLOSE / Méthode d'Accès	2 - 8
g) Interface Méthode d'Accès / Noyau (MES)	2 - 8
h) Interface FMS-E / SSP FMS	2 - 8
i) Interface FMS-E / SUP FMS	2 - 9
2.2.3. - Les Tables du Système de Fichiers	2 - 10
a) Architecture des tables	2 - 10
b) Les zones de travail de FMS-E	2 - 12
c) La FAU : Le Descripteur d'une Unité d'Accès à un Fichier	2 - 13
d) Le WCB	2 - 19
e) Le Sémaphore de FAU publique	2 - 28
f) Le DF : Le Descripteur de Fichier	2 - 29
g) Le DFU : Le descripteur de FU	2 - 36
2.2.4. - Interface : Superviseur / FMS	2 - 39
a) Paramètres d'entrée	2 - 40
b) Paramètres de sortie	2 - 41
2.2.5. - Interface FMS-E / SUPFMS	2 - 43
a) interface FMS-E (OPEN-CLOSE) / ANASFU	2 - 44
b) Généralités sur l'allocateur de pavés	2 - 45
c) Interface FMS-E / GETP	2 - 46
d) Interface FMS-E / FREEP	2 - 47
e) Interface FMS-E / PVIN	2 - 48
f) Interface FMS-E / PVOU	2 - 49
2.2.6. - Interface FMS-E / SSPFMS	2 - 50
a) Interface FMS-E / VERFCB	2 - 51
b) Interface FMS-E / RADIX	2 - 52
c) Interface FMS-E / VERBUF	2 - 53

d) Interface FMS-E / PARBUF	2 - 5 4
e) Interface FMS-E / BIDON	2 - 5 5
f) Interface FMS / sous-programmes d'adressage 1024 K	2 - 5 6
2.3. - LE NOYAU DE FMS-E	2 - 5 7
2.3.1. - Schéma général du Noyau KERADR	2 - 5 7
2.3.2. - Les primitives du Noyau KERADR	2 - 5 8
a) Le module MDK	2 - 5 8
b) Le module MOC	2 - 5 9
c) Le module MES	2 - 5 9
d) Le module ESB	2 - 5 9
2.3.3. - Interface OPEN-CLOSE / Noyau (MOC)	2 - 6 1
2.3.4. - Interface Méthode d'Accès / Noyau (MES)	2 - 6 2
2.3.5. - Interface Méthode d'Accès / ESBFMS	2 - 6 3
2.3.6. - Interface MES, ESB / MDK	2 - 6 4
2.3.7. - Interface MOC / MDK	2 - 6 5
2.4. - LE MODULE OPEN-CLOSE	2 - 6 6
2.4.1. - Schéma général	2 - 6 6
2.4.2. - Les primitives du module OPEN-CLOSE	2 - 6 7
a) Les primitives OPEN-CLOSE	2 - 6 7
b) Les primitives système	2 - 6 9
c) Codage de PP (Primitive Précédente)	2 - 8 0
2.4.3. - Interface FMS / Méthode d'Accès	2 - 8 2
2.4.4. - interface OPEN-CLOSE / Méthode d'Accès	2 - 8 3
2.4.5. - Interface FMS / OPEN-CLOSE	2 - 8 4
2.5. - UNE METHODE D'ACCES	2 - 8 5
2.5.1. - Tableau général d'identification des Méthodes d'Accès	2 - 8 6
2.5.2. - Structure programme d'une Méthode d'Accès	2 - 8 7
2.5.3. - Le Séquentiel	2 - 9 0
a) Séquentiel Pur	2 - 9 1
b) Séquentiel Pur Statique	2 - 9 2
c) Portion d'Article Statique (PAS)	2 - 9 4
d) Portion d'Article Dynamique (PAD)	2 - 9 7
e) Commandes du Séquentiel	2 - 1 0 0

### **3 - CONSEILS D'UTILISATION**

<b>3.1 - CONSEILS GENERAUX</b>	<b>3 - 2</b>
3.1.1. - La taille des granules	3 - 2
<b>3.2 - UTILISATION DES METHODES D'ACCES</b>	<b>3 - 5</b>
3.2.1. - Le séquentiel indexé	3 - 5

### **ANNEXE**

<b>A : SYNOPTIQUE</b>	<b>A - 1</b>
Les comptes rendus de FMS	A - 1
Correspondance primitive compte-rendu	A - 2
FCB	A - 4
Valeurs limites de FMS - FUP	A - 11
Taille des fichiers	A - 15
Le partage des fichiers	A - 16
Schémas de FMS	A - 20
les tables de FMS	A - 25
La FAU	A - 26
Le WCB	A - 27
Le DF	A - 29
La DFU	A - 30
Le Sémaphore de FAU Publique	A - 32
Tableau des méthodes d'Accès	A - 33

## 1 - GENERATION

### 1.1 - PRESENTATION

Le système de fichiers FMS se présente sous plusieurs versions. Il appartient à l'utilisateur de choisir la version qu'il désire utiliser, et de configurer chaque version en fonction des besoins de son application.

#### FMS-E

FMS-E est un système de fichiers à la carte. C'est-à-dire qu'il appartient à l'utilisateur de choisir, parmi l'ensemble qui lui est proposé, les modules adaptés aux besoins de son application.

FMS-E possède une structure modulaire complète, ce qui permet très facilement à un utilisateur, d'adapter la taille du Système de Fichiers aux services souhaités.

Ce chapitre a deux buts :

- Aider l'utilisateur à choisir un ensemble de modules constituant une version opérationnelle de FMS-E.
- Fournir les règles pratiques permettant à un utilisateur, de construire une version opérationnelle de FMS-E, et de l'intégrer dans un système d'Exploitation.

#### FMS-G

FMS-G est un sur-ensemble de FMS-E. Son architecture logicielle est celle de FMS-E et les nouvelles fonctionnalités sont apportées par l'existence d'un noyau spécial KERGDI sachant gérer des petits disques (5 M octets) et des grands-disques (50 M octets et plus).

#### GENFMS

GENFMS permet de générer les différentes versions de FMS.



## 1.2 - DESCRIPTION DU PRODUIT FMS

FMS est un système de fichiers totalement résident, réentrant pour de 1 à 128 tâches software et utilisable sur un calculateur de taille mémoire, jusqu'à 1024 K mots.

FMS est composé de :

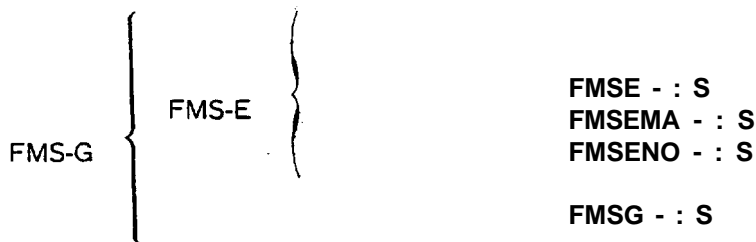
Un ensemble de bibliothèques, de modules link-éditables :

- FMSE - : S : bibliothèque performance, contenant les 3 méthodes d'accès standard avec les options de performance.
- FMSEMA - : S : bibliothèque méthodes d'accès, contenant les méthodes d'accès complémentaires : Direct à trous, Séquentiel indexé,...
- FMSENO - : S : bibliothèque contenant les extensions Noyau et Openclose.
- FMSG - : S : bibliothèque contenant le noyau de gestion des grands disques.

Un outil de génération automatique :

- GENFMS : symbolique MACP produisant de l'assembleur.

Les deux versions de FMS



### 1.2.1 - Composition des bibliothèques

Remarque :

Les informations de taille des modules programmes fournies dans ce manuel ne seront pas mises à jour. Elles correspondent à la version FMS IE 51 de Mai 1982 et sont arrondis à la valeur supérieure.

a) Bibliothèque Performance : FMSE - : S

- IDPRAP	: article identification	
- KERADR	: noyau rapide, contient - l'accès direct rapide - la NON relecture de contrôle	2750
- ESBFMS	: module de bufférisation	475
- OPCGEN	: OPEN-CLOSE général, contient - la bufférisation - les méthodes d'accès avec buffer de travail (Seq. Indexé, Seq. chaîné) - la NON relecture de contrôle - la gestion de volume	1700
- SEQBUF	: séquentiel bufférisé	460
- INDBUF	: indexé bufférisé	1200
- DIRBUF	: direct bufférisé et contient l'accès séquentiel bufférisé après DSEL.	225

Total env. = 6810

b) Bibliothèque Méthodes d'Accès : FMSEMA - : S

- IDPEMA	: article identification	
- DITFMS	: direct à trous, remplace DIRBUF et contient la gestion de l'allocation des articles dans le Fichier, et la gestion de 2 milliards d'articles	800
- SIXFMS	: Séquentiel Indexé	3350
- SCHFMS	: Séquentiel Chaîné	800
- DIVFMS	: Direct Longueur Variable	1000

Total env. = 5950

c) Bibliothèque Extension Noyau : FMSENO - : S

- IDPEN0	: article identification	
- OPCPUB	: OPEN-CLOSE gérant les FAU Publiques, remplace OPCGEN	2700
- SPCRFU	: Sous-programme de gestion de volume (350) intégré dans le superviseur	

Total env. = 2700

d) Bibliothèque FMSG - : S

- IDPFMG	article identification	
- KERGDI	noyau grand disques, remplace KERADR et permet avec le DIT la gestion de 2 milliards d'articles.	4250

Total env. = 4250

1.2.2 - Règle générale de construction

Une version opérationnelle de FMS-E ou FMS-G doit être constituée de :

- Au minimum
  - Un Noyau (1 seul)
  - Un OPEN-CLOSE (1 seul)
  - Une méthode d'accès
- A cela on peut ajouter
  - des méthodes d'accès
    - ajout : exemple, ajouter le Séquentiel Indexé.
    - remplacement : exemple, le Direct à Trous remplace le Direct Bufferisé.

- des options de performance

remplacement : exemple  
les FAU publiques OPCPUB remplace OPCGEN.

### 1.2.3 - Les possibilités de choix : FMS-E et FMS-G

Le choix de l'utilisateur porte sur deux ensembles différents.

#### a) Le choix des méthodes d'accès

- Séquentiel (%SEQ)  
Le séquentiel permet de traiter
  - les fichiers Séquentiel
  - tout fichier en accès séquentiel
  - la portion d'article (Indexé, Direct,...)
- Indexé (%IND)
- Direct (%DIR) ou (exclusif) Direct à trous (%DIT)  
(Le Direct à Trous est un sur-ensemble du Direct)
- Séquentiel indexé (%SIX)
- Séquentiel chaîné (%SCH)
- Direct Longueur Variable (%DIV)

#### b) Le choix des options de performance

- Les FAU Publiques (%FNUMP)  
Le choix de ce service entraîne le choix du module suivant :
  - OPCPUB : L'OPEN-CLOSE FAU publiques
- La gestion de volume : (%FNUMP)  
Le choix de ce service est fait à la génération du superviseur, il est intégré en overlay dans le Système d'Exploitation. Pour FMS ce service est rendu dans la mesure ou le système choisi contient OPCGEN ou OPCPUB.

**c) Pour FMS-G**

- La gestion des grands disques (et des petits).  
Ce service est rendu dans la mesure où le système choisi contient :
  - KERGDI : Le Noyau Grands Disques (%GDI)
  - La gestion de volume.
  
- Les fichiers Direct de 2 milliards d'articles.  
Ce service est rendu dans la mesure où le système choisi contient :
  - KERGDI : Le Noyau Grands Disques (%GDI)
  - DITFMS : Le Direct à Trous (%DIT)
  - La gestion de volume.

**d) Rappel :**

Les modules suivants sont emboîtés du point de vue fonctionnel :

- KERGDI > KERADR
- OPCPUB > OPCGEN
- DITFMS > DIRBUF

#### 1.2.4 - Exemples de choix élémentaires FMS-E

La liste suivante indique pour un service choisi et un seul quels modules sont nécessaires. Pour le choix d'un ensemble de services, il faut faire l'union des modules nécessaires (ne pas compter 2 fois le même module).

##### Séquentiel Bufferisé %SEQ

- KERADR : Noyau Rapide
- ESBFMS : Module de Bufferisation
- OPCGEN : OPEN-CLOSE Général
- SEQBUF : Séquentiel Bufferisé

##### Indexé Bufferisé %SEQ %IND

- KERADR : Noyau Rapide
- E S B F M S : Module de Bufferisation
- OPCGEN : OPEN-CLOSE Général
- S E Q B U F : Séquentiel Bufferisé
- I N D B U F : Indexé Bufferisé

- Ce système permet le traitement :

- des fichiers Séquentiel avec bufferisation
- des fichiers Indexé avec Portion d'Article Bufferisé.

##### Direct Rapide %DIR

- KERADR : Noyau Rapide
- OPCGEN : OPEN-CLOSE Général
- D I R B U F : Direct Bufferisé
- E S B F M S : Module de bufferisation

##### Direct à Trous (Rapide) %DIT

- KERADR : Noyau Rapide
- OPCGEN : OPEN-CLOSE Général
- D I T F M S : Direct à Trous
- E S B F M S : Module de bufferisation

##### Séquentiel Indexé (rapide) %EMA %SIX

- KERADR : Noyau Rapide
- OPCGEN : OPEN-CLOSE Général
- S I X F M S : Séquentiel Indexé

##### Séquentiel Chaîné (rapide) %EMA %SCH

- KERADR : Noyau Rapide
- O P C G E N : OPEN-CLOSE Général
- S C H F M S : Séquentiel Chaîné

La liste des exemples ci-dessus est donnée à titre d'information, pour comprendre quelle version opérationnelle sera construite par l'outil de génération automatique GENFMS.

En effet GENFMS ne demande à l'utilisateur qu'un ensemble de commandes simples, à partir desquelles il construit automatiquement une version opérationnelle de FMS-E en respectant les règles de combinaison de modules.

**GENFMS propose 1 "menu"**

**%FMSE : Commande qui génère la version Performance avec les 3 méthodes d'accès standard.**

- KERADR
- ESBFMS
- OPCGEN
- SEQBUF
- INDBUF
- DIRBUF

**GENFMS propose FMS-E à la carte %EMA %SEQ %DIT %SIX**

**%ALACARTE : Cet ensemble de commandes génère :**

- KERADR
- ESBFMS
- OPCGEN
- SEQBUF
- DITFMS
- SIXFMS

**Cette version opérationnelle permet de traiter :**

- des fichiers Séquentiel avec bufferisation
- des fichiers Direct rapide et Direct à Trous rapide avec la portion d'article
- des fichiers Séquentiel Indexé rapide

### c) Principe de génération

Le module Table puis les modules programmes doivent être "link-édités" avec le Système d'Exploitation.

La construction effective d'une version opérationnelle de FMS se passe donc, pendant la phase de génération du Système d'Exploitation.

Edition de Liens de

- TABFMS - : S
- Les modules programmes
- IOCS
- Les drivers
- etc . . .

L'utilisation de GENFMS constitue donc une phase préparatoire ayant pour but de produire un fichier :

- TABFMS - : S
- FCFMSE - : S

Pendant cette phase préparatoire l'utilisateur réalise donc le choix de sa version opérationnelle.

- Choix des modules programmes
  - choix des méthodes d'accès
  - choix des options de performance

Ce choix est mémorisé par :

- les références externes de TABFMS - : S.
  
- Choix des FU gérées par FMS ou des PU (FU initiale) gérée par FMS  
Ce choix est mémorisé par la TDFU de TABFMS - : S.

### 1.3 - GENERATION DE FMS

#### 1.3.1 - Principe général

Une version opérationnelle de FMS est fabriquée à partir de :

- Des modules programmes
- Un module Table : TABFMS - : S

##### a) Des modules programmes

Un ensemble de modules "Link-éditable" correctement choisi, à l'aide de GENFMS dans les bibliothèques de FMS. Voir chapitre précédent (1.2.).

##### b) Un module Table : TABFMS - : S

Ce module "Link-éditable" n'est pas fourni par les bibliothèques de FMS mais doit être construit par l'utilisateur à l'aide de GENFMS. Il contient :

- 1) TDFU : La Table des FU qui devront être gérées par cette version opérationnelle de FMS.

Sur les disques avec gestion de volume l'utilisateur définit pour chaque unité physique (PU) ou emplacement de PU, la FU initiale gérée par FMS ainsi que le nombre maximal de granules gérables pour cette PU.

1 < ou = PUNBGRAN < ou = 32000 Organisation Standard

1 < ou = PUNBGRAN < ou = 32767 Organisation Grand disque.

Sur les disques sans gestion de volume l'utilisateur définit de façon statique l'ensemble des FU gérées par FMS pour chaque PU. Il définit pour chaque FU :

- Le Nom de la FU (ex : FU = D2)

FU = D2 à DF, E1 à EB, ED à EF

- Le nombre de granules maximum que FMS devra être capable de gérer sur le ou les supports adressables par cette FU.

exemple : NBGRAN = 1197

Règles

2 < ou = NBGRAN < ou = 2000 Organisation Standard

1 < ou = NBGRAN < ou = 32656 Organisation Grand Disque.

- Lorsque l'utilisateur ne possède pas d'informations sur la taille des granules, il pourra calculer NBGRAN à partir de la taille standard TG = 16 secteurs. Voir chapitre 3.1, et Manuel de Référence FUP : FUP4 : FUINI.

Au niveau du ou des supports adressables par cette FU, l'utilisateur pourra alors choisir (FUP4 : FUINI) des tailles de granules supérieure ou égale à 2 K mots.

- 2) TABMA, TOPCMA : Deux tables d'aiguillage pour l'appel des méthodes d'accès par :

- Le tronc commun de FMS
- Le module OPEN-CLOSE

- 3) Des références externes ayant pour but l'édition de liens automatiques (LLNK) des modules de bibliothèque correspondant au choix de l'utilisateur.

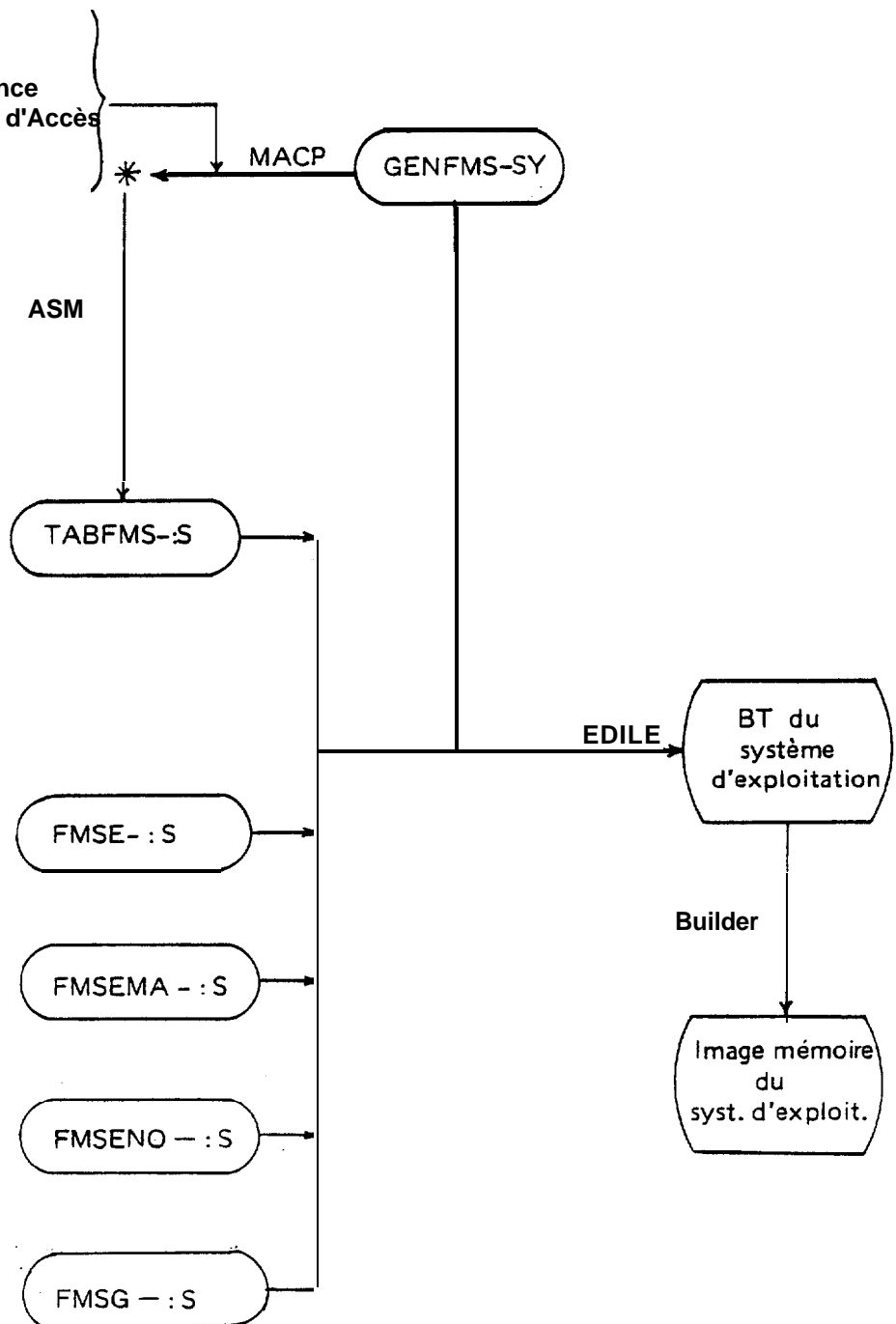


### 1.3.2 - Schéma de génération

Macro-instructions :

- FU, NBGRAN
- Options de performance
- Extension Méthodes d'Accès
- Méthodes d'Accès
- Extension Noyau

Les bibliothèques  
de FMSE



### 1.3.3 - Exemple de production de TABFMS

#### a) Phase de macro-définition

```
* CALL MACP
* SI GENFMS - : S, D2
* SO ZE
* BO ZE
* LO ZE
* IMAC                               lecture de GENFMS
```

#### b) Phase de macro-génération de TABFMS

```
* SI CR                               macro-instructions sur cartes
* SO TABFMS - SY, D2  Symbolique assembleur
* CMAC
```

#### c) Génération de la TDFU

disque sans reconfiguration dynamique : génération statique

```
%CONFMS FU = D2  NBGRAN= 726
%CONFMS FU = D4  NBGRAN= 190
%CONFMS FU = D8  NBGRAN= 672
```

cartouche (Volume monté avec D3) : génération dynamique préinitialisée

```
%PUFMS FUI = D9  OFU= 0 NBFMSMAX = 3 PUNBGRAN = 3500
%CONFMS FU = D3  NBGRAN = 1197 [OFU = 0]
```

disque pack (volume démonté) : génération dynamique standard

```
%PUFMS FUI = E5  OFU= 1 NBFMSMAX = 3 PUNBGRAN = 30 000
```

#### d) Choix à la carte du système de fichiers

Choix des options de performance

FAU publiques

```
%FNUMP FNUMPI = 32 FNUMP2 = 64 FNUMP3 = 248 FNUMP4 = 255
%GDI                               Noyau grand disque
```

Choix des méthodes d'accès

```
%SEQ                               Séquentiel
%IND                               Indexé
%DIR ou %DIT                       Direct ou Direct à Trous
%SIX                               Séquentiel Indexé
%SCH                               Séquentiel Chaîné
%DIV                               Direct Variable
```

### Désignation des bibliothèques

%EMA	si DIT, SIX, SCH, DIV
%ENO	si FNUMP
%FMG	si GDI
% A LA CARTE	obligatoire fin du choix à la carte
* END	

### e Phase d'assemblage de TABFMS

- \* CALL ASM
- \* SI TABFMS - SY, D2      Symbolique généré ci-dessus
- \* BO TABFMS - : S, D2      MOL pour la génération du Système
- \* SO ZE
- \* LO LP
- \* IASM
- \* CLOSE BO

#### 1.3.4 - Règles de fabrication de TABFMS

Les règles suivantes sont indiquées en reprenant point par point l'exemple ci-dessus.

### a Phase de macro-définition

- \* SI nomfic - cat, FU doit désigner le fichier contenant le symbolique de GENFMS

### b Phase de génération de TABFMS

- \* SI CR OU \* SI TK OU \* SI nomfic - cat, FU selon le support d'entrée des macro-instructions (%).

### c Génération de la TDFU

#### • Génération statique

Les macros % CONFMS qui correspond à un ou des disques sans gestion de volume, ou avec gestion d'espace mais sans reconfiguration dynamique doivent se trouver en tête en particulier pour le disque système sous BOS-D. La génération est dite statique. Implicitement OFU est initialisé à 0 signifiant organisation standard.

#### • Génération dynamique

Macro %PUFMS FUI = D9 OFU = O NBFMSMAX = 3 PUNBGRAN = 3500  
[LSEC = 128 NZUEP= O]

- FUI désigne le nom de la fu initiale c'est-à-dire la première déclarée dans les macro-instructions de GENIO pour cet emplacement d'Unité Physique. Son nom est choisi parmi les 28 FU géables par FMS D2 à DF, E1 à EB, ED, EF.

- **OFU** désigne le type d'organisation utilisé pour les espaces des disques montés sur cet emplacement physique.  
Lorsque l'emplacement n'est pas totalement prédéfini par des macros %CONFMS :  
OFU doit être égal à 1 si au moins une des fu de cet emplacement, est gérée avec l'organisation Grand Disque.  
OFU = 0 Organisation Standard  
OFU = 1 Organisation Grand Disque  
Il est rappelé que l'organisation Grand Disque n'est gérée par FMS-G qu'avec les systèmes possédant la gestion de volume et sur les disques pour lequel le montage de volume fonctionne.

- **NBFMSMAX** désigne le nombre maximal d'espaces gérés par FMS pour cette unité physique, c'est-à-dire sur un même volume susceptible d'être monté sur cet emplacement de PU.

$$1 \leq \text{NBFMSMAX} \leq 16.$$

- **PUNBGRAN** désigne le nombre maximal de granules que FMS devra gérer sur cet emplacement de PU pour un même volume.

$$1 \leq \text{PUNBGRAN} \leq 32767 \quad \text{si OFU} = 1$$

$$1 \leq \text{PUNBGRAN} \leq 2000 \quad \text{si OFU} = 0$$

Remarque :

Pour une génération dynamique préinitialisée, la taille de la table FMS (DPU) est définie, soit par la macro %PUFMS soit par la somme des macros %CONFMS de préinitialisation, à savoir la plus grande de ces deux valeurs.

#### Exemple de choix de NBFMSMAX et PUNBGRAN

Soit trois cartouches a, b, c utilisables sur la même unité physique.

Ⓐ 1 espace

NBG = 1000
------------

Ⓑ 2 espaces

NBG = 700	NBG = 700
-----------	-----------

Ⓒ 3 espaces

NBG = 400	NBG = 400	NBG = 400
-----------	-----------	-----------

Dans cet exemple NBFMSMAX = 3 PUNBGRAN = 1400 dans la macro %PUFMS

- **LSEC** désigne la taille secteur des disques associés à cet emplacement de PU.

$$128 \leq \text{LSEC} \leq 2048 \quad \text{avec}$$

$$2^i = \text{LSEC} \quad \text{et } i = [7, 8, 9, 10, 11]$$

- **NZUEP** désigne le numéro de zone de l'allocateur de pavés système qui contient des secteurs de la taille correspondante.

NZUEP = 0      zone ZUEP avec 128 mots  
NZUEP = 6      zone ZUEP6 configurable  
NZUEP = 7      zone ZUEP7 configurable dans RTES-D seulement.

Remarque :

Actuellement la taille des secteurs disque est figée à 128 mots. Par ailleurs l'utilisation des ZUEP 6 et 7 permet de choisir pour chaque PU la zone de pavés qui sera utilisée. Cela permet de répartir les ressources ZUEP sur un ensemble de PU à fin d'optimiser le temps de réponse. Par exemple de réserver une ZUEP à une seule PU à fin de garantir le meilleur temps d'accès à cette PU.

Macro %CONFMS FU = D3 NBGRAN = 1197 [OFU = 0]

- FU désigne une FU qui sera générée de façon statique dans la TDFU. Cette FU sera accessible après l'INIT système. Si la FU préinitialise un emplacement de PU avec montage de volume elle sera détruite par la reconfiguration dynamique des commandes de montage de volume. Son nom est choisi parmi les 28 FU gérables par FMS DR à DF, E1 à EB, ED, EF.
- NBGRAN définit le nombre maximal de granules gérables par 'FMS' sur cette FU.

$2 \leq \text{NBGRAN} \leq 2000$       si OFU = 0  
 $1 \leq \text{NBGRAN} \leq 32656$       si OFU = 1

- OFU = 0      signifie qu'il faut générer un descripteur de FU dont la taille sera adaptée à l'organisation standard des espaces disques.
- OFU = 1      signifie que la taille sera adaptée à l'organisation grand disque des espaces disques (FMS-G).

Si OFU est absent, la valeur prise par défaut est celle du paramètre OFU de la macro PUFMS précédente, ou zéro si cette macro n'existe pas.

**d** Choix du Système de fichier

% A LA CARTE      Choix à la carte  
\* END

L'utilisateur choisit dans la liste indiquée les macros qui correspondent à son choix, en respectant les règles d'exclusion ou de citation de bibliothèques.

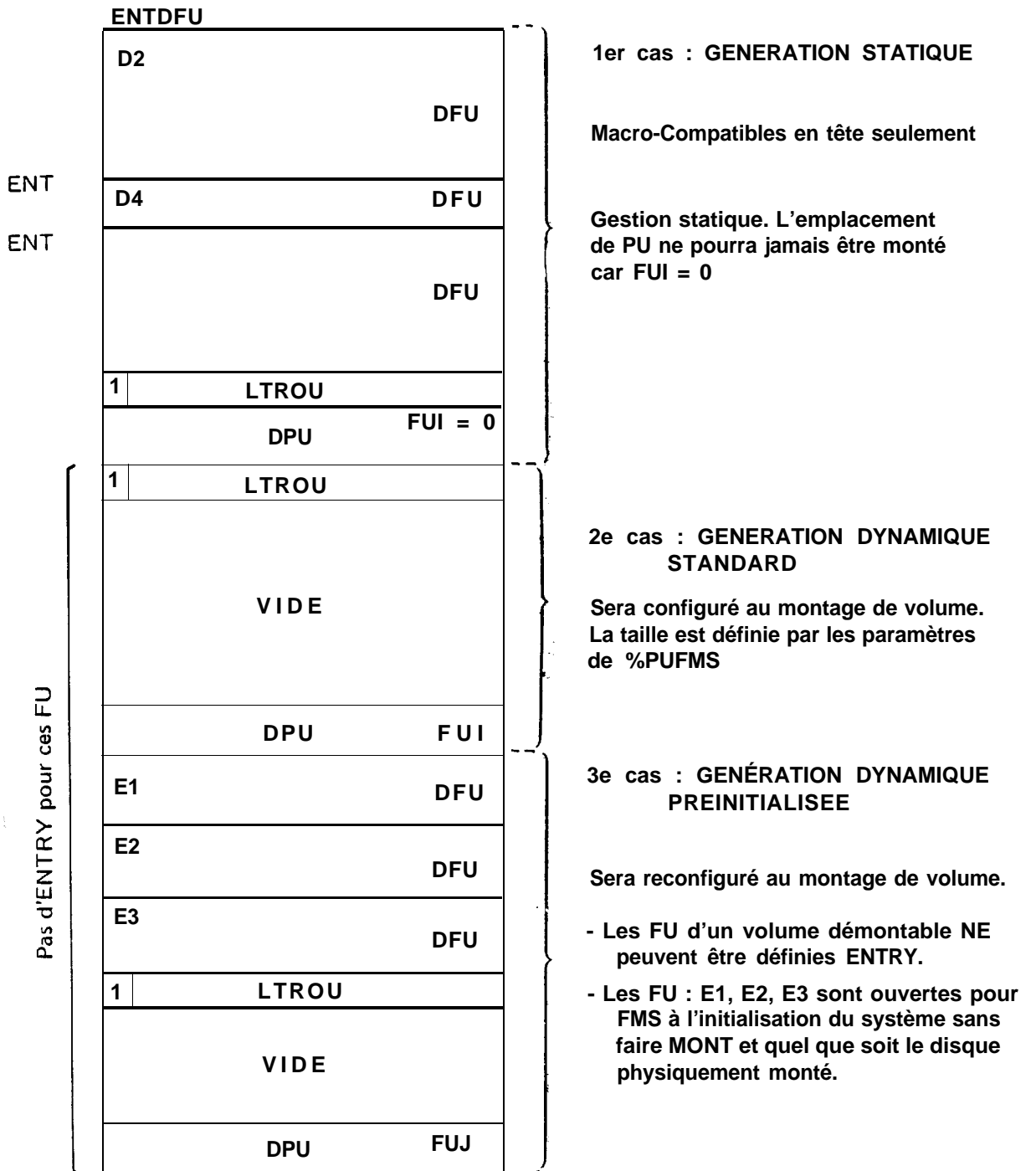
%FNUMP      (Utilisable seulement avec % A LA CARTE ou %TDFU)

Règle  $0 \leq \text{FNUMP1} \leq \text{FNUMP2} \leq \text{FNUMP3} \leq \text{FNUMP4} \leq 255$   
ou  $\text{FNUMP3} = 0$  et  $\text{FNUMP4} = 0$

$[\text{FNUMP1}, \text{FNUMP2}]$  et  $[\text{FNUMP3}, \text{FNUMP4}]$   
définissent deux plages de FNUP réservées aux FAU Publiques.

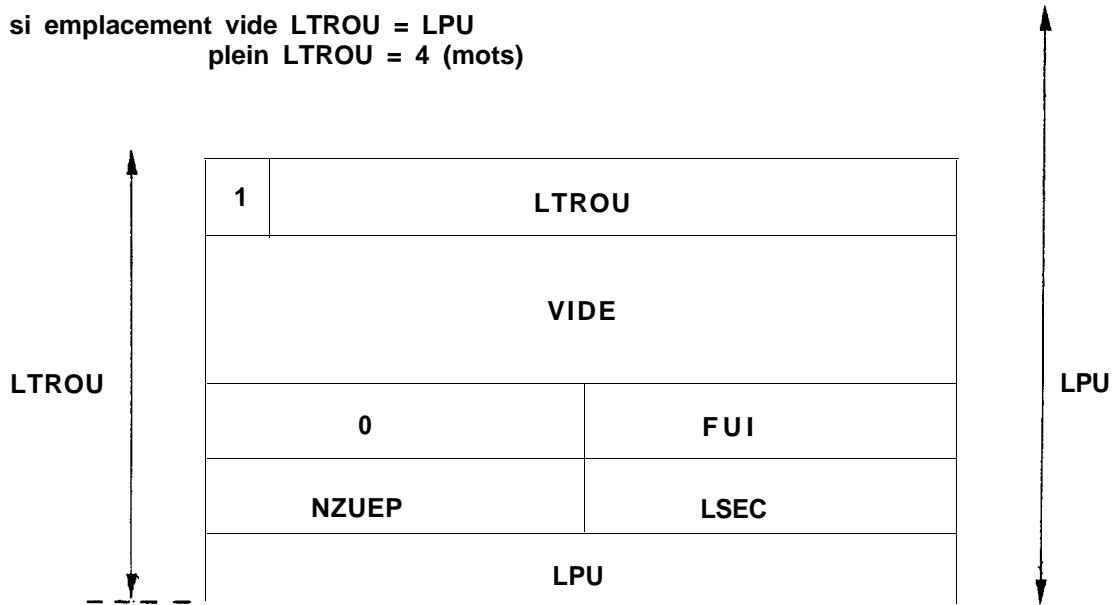
1.3.5 - Génération de la TDFU

a) T D F U

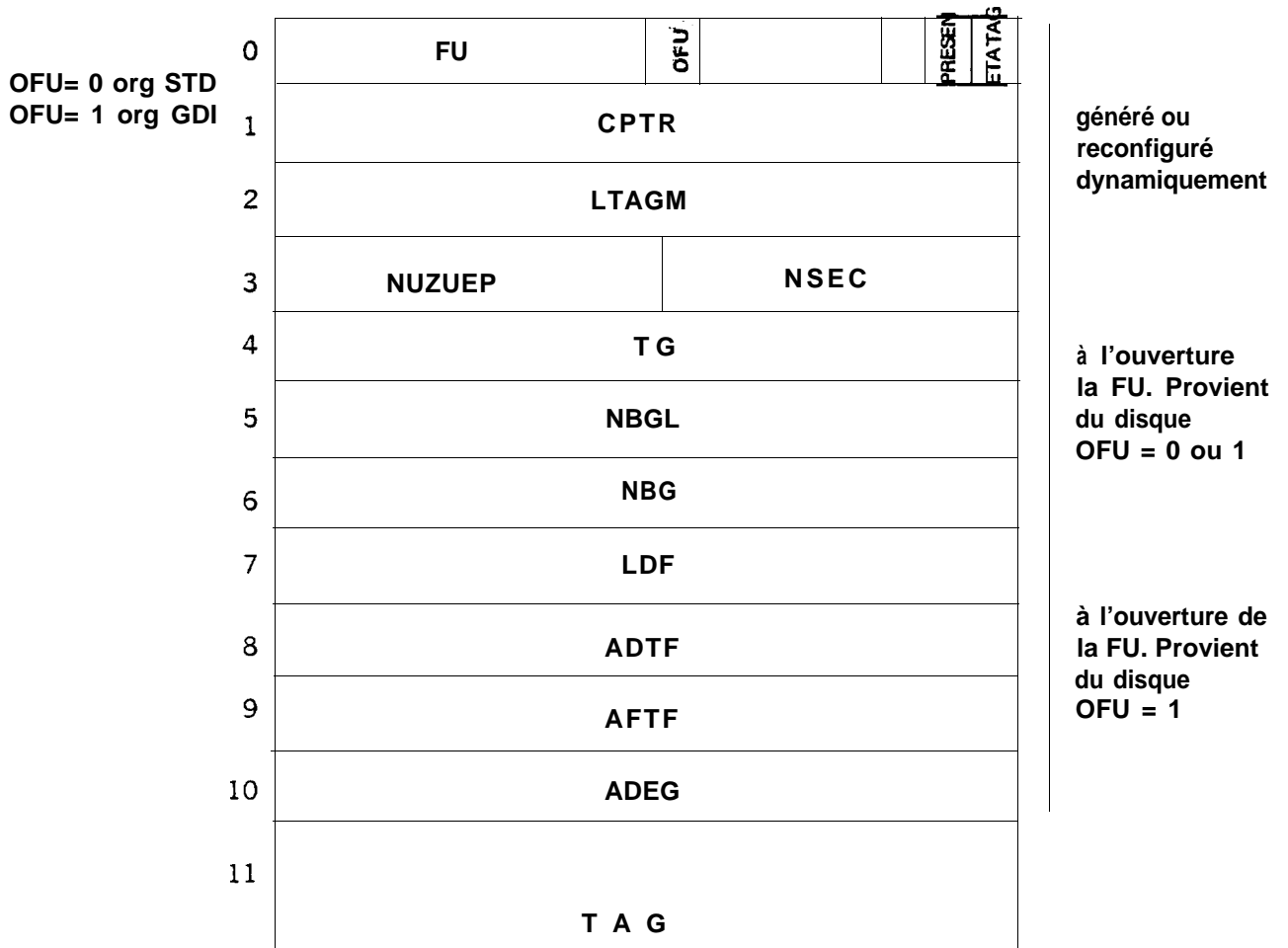


b) DPU : Descripteur de PU

si emplacement vide LTRou = LPU  
plein LTRou = 4 (mots)



c) DFU : Descripteur de FU



d) Fabrication de la TDFU

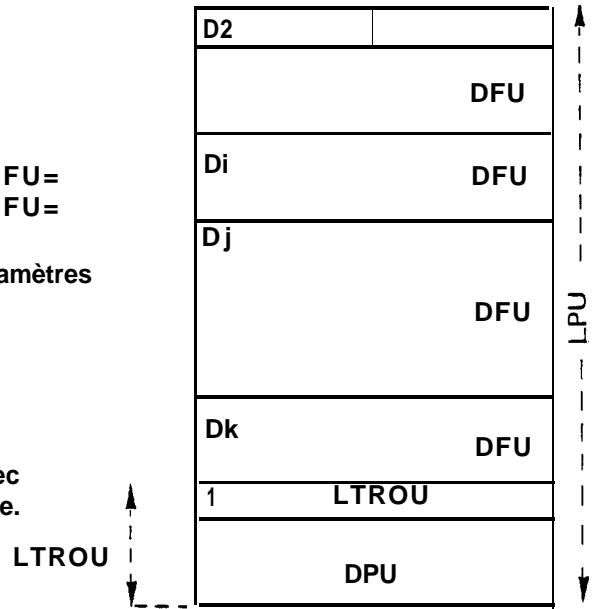
1°) Génération statique

```
%CONFMS    FU = D2    NBGRAN =
% C O N F M S "        "
% C O N F M S "        "
% C O N F M S "        "        OFU=
% C O N F M S "        "        OFU=
```

Remarque : la longueur LPU est définie par les paramètres NBGRAN et OFU des macros %CONFMS.

Utilisation :

Disques à têtes fixes,  
disques souples,  
disques D1, D2 sous BOS-D c'est-à-dire disques avec table d'espace mais sans reconfiguration dynamique.



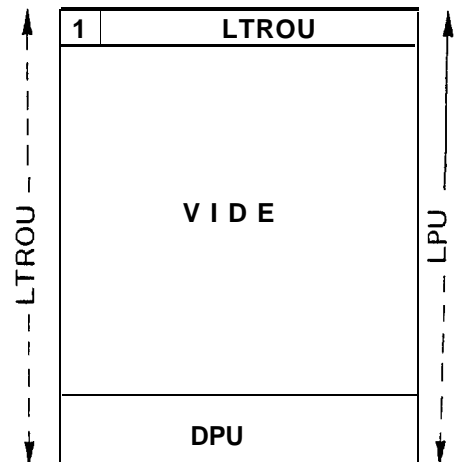
2°) Génération dynamique standard (vide)

```
%PUFMS    FUI=    OFU=    NBFMSMAX=    PUNBGRAN=    [ LSEC=    NZUEP=    ]
```

Remarque : la longueur LPU est définie par les paramètres OFU, NBFMSMAX et PUNBGRAN de la macro %PUFMS.

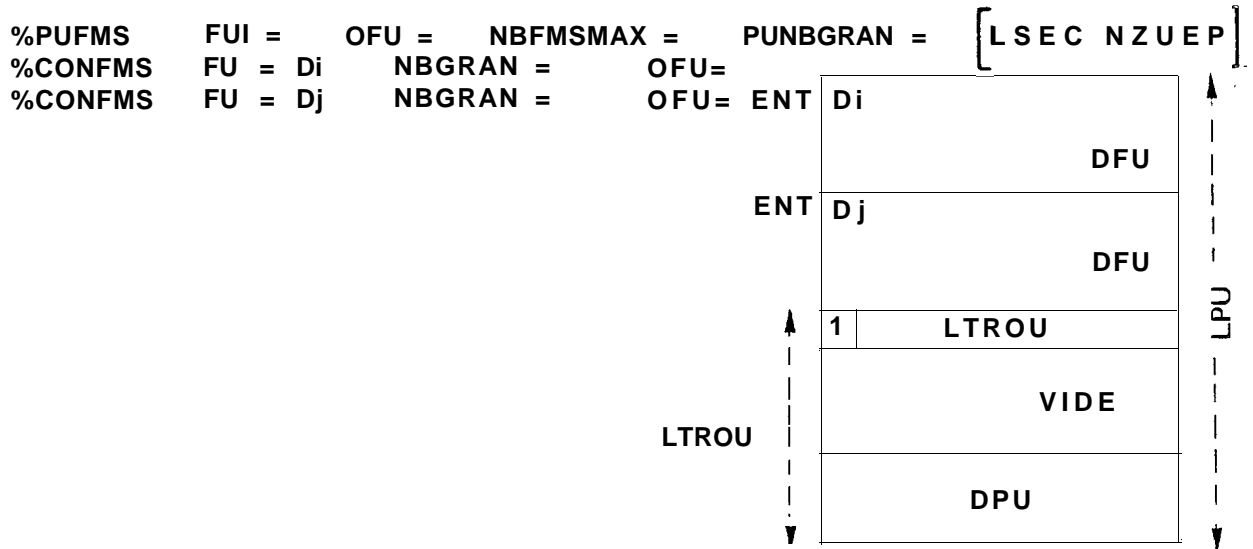
Utilisation :

Cas standard de gestion de volume. Cas des disques pour lesquels la reconfiguration dynamique est faite automatiquement au lancement du système.  
Ex : disques systèmes de MPES.





3°) Génération dynamique préinitialisée



La taille LPU est définie par la plus grande des deux valeurs :

- LPU1 = f (OFU, NBFMSMAX, PUNBGRAN) de la macro %PUFMS
- LPU2 = f (NBGRAN, OFU) de toutes les macros %CONFMS.

Dans l'exemple, la présence de la zone vide indique que LPU1 (%PUFMS) est plus grand que LPU2.

Utilisation : Cas spécial statique et dynamique.

Exemple : Macro compatibles BOS-D, TSM pour le disque D1, D2 :

- . Statique pour BOS-D
- . Dynamique pour TSM.

<b>2 - STRUCTURE INTERNE FMS s E</b>	<b>2 - 1</b>
<b>2.1. - PRESENTATION</b>	<b>2 - 2</b>
2.1.1. - Schéma d'un Système d'Exploitation	2 - 2
2.1.2. - FMS-E et les éléments Link-édités	2 - 3
2.1.3. - Description générale des modules	2 - 4
a) Le Superviseur	2 - 4
b) Les modules de FMS-E	2 - 4
<b>2.2. - DESCRIPTION GENERALE</b>	<b>2 - 6</b>
2.2.1. - Environnement d'une Méthode d'Accès dans FMS-E	2 - 6
2.2.2. - Description générale des interfaces	2 - 7
a) Interface Superviseur / FMS	2 - 7
b) Interface FMS / OPEN-CLOSE	2 - 7
c) Interface OPEN-CLOSE / Noyau (MOC)	2 - 7
d) Interface Noyau (MDK) / IOCS	2 - 7
e) Interface FMS / Méthode d'Accès	2 - 7
f) Interface OPEN-CLOSE / Méthode d'Accès	2 - 8
g) Interface Méthode d'Accès / Noyau (MES)	2 - 8
h) Interface FMS-E / SSP FMS	2 - 8
i) Interface FMS-E / SUP FMS	2 - 9
2.2.3. - Les Tables du Système de Fichiers	2 - 10
a) Architecture des tables	2 - 10
b) Les zones de travail de FMS-E	2 - 12
c) La FAU : Le Descripteur d'une Unité d'Accès à un Fichier	2 - 13
d) Le WCB	2 - 19
e) Le Sémaphore de FAU Publique	2 - 28
f) Le DF : Le Descripteur de Fichier	2 - 29
g) Le DFU : Le Descripteur de FU	2 - 36
2.2.4. - Interface : Superviseur / FMS	2 - 39
a) Paramètres d'entrée	2 - 40
b) Paramètres de sortie	2 - 41
2.2.5. - Interface FMS-E / SUPFMS	2 - 43
a) Interface FMS-E (OPEN-CLOSE) / ANASFU	2 - 44
b) Généralités sur l'allocateur de pavés	2 - 45
c) Interface FMS-E / GETP	2 - 46
d) Interface FMS-E / FREEP	2 - 47
e) Interface FMS-E / PVIN	2 - 48
f) Interface FMS-E / PVOU	2 - 49
2.2.6. - Interface FMS-E / SSPFMS	2 - 50
a) Interface FMS-E / VERFCB	2 - 51
b) Interface FMS-E / RADIX	2 - 52
c) Interface FMS-E / VERBUF	2 - 53

d)	Interface FMS-E / PARBUF	2 - 5 4
e)	Interface FMS / BIDON	2 - 5 5
f)	Interface FMS/ Sous programmes d'adressage 1024 K	2 - 5 6
2.3.	- LE NOYAU DE FMS-E	2 - 5 7
2.3.1.	- Schéma général du Noyau KERADR	2 - 5 7
2.3.2.	- Les primitives du Noyau KERADR	2 - 5 8
a)	Le module MDK	2 - 5 8
b)	Le module MOC	2 - 5 9
c)	Le module MES	2 - 5 9
d)	Le module ESB	2 - 5 9
2.3.3.	- Interface OPEN-CLOSE / Noyau (MOC)	2 - 6 1
2.3.4.	- Interface Méthode d'Accès / Noyau (MES)	2 - 6 2
2.3.5.	- Interface Méthode d'Accès / ESBFMS	2 - 6 3
2.3.6.	- Interface MES, ESB / MDK	2 - 6 4
2.3.7.	- Interface MOC / MDK	2 - 6 5
2.4.	- LE MODULE OPEN-CLOSE	2 - 6 6
2.4.1.	- Schéma général	2 - 6 6
2.4.2.	- Les primitives du module OPEN-CLOSE	2 - 6 7
a)	Les primitives OPEN-CLOSE	2 - 6 7
b)	Les primitives système	2 - 6 9
c)	Codage de PP (Primitive Précédente)	2 - 8 0
2.4.3.	- Interface FMS / Méthode d'Accès	2 - 8 2
2.4.4.	- Interface OPEN-CLOSE / Méthode d'Accès	2 - 8 3
2.4.5.	- Interface FMS / OPEN-CLOSE	2 - 8 4
2.5.	- UNE METHODE D'ACCES	2 - 8 5
2.5.1.	- Tableau général d'identification des Méthodes d'Accès	2 - 8 6
2.5.2.	- Structure programme d'une Méthode d'Accès	2 - 8 7
2.5.3.	- Le Séquentiel	2 - 9 0
a)	Séquentiel Pur	2 - 9 1
b)	Séquentiel Pur Statique	2 - 9 2
c)	Portion d'Article Statique (PAS)	2 - 9 4
d)	Portion d'Article Dynamique (PAD)	2 - 9 7
e)	Commandes du Séquentiel	2 - 10 0

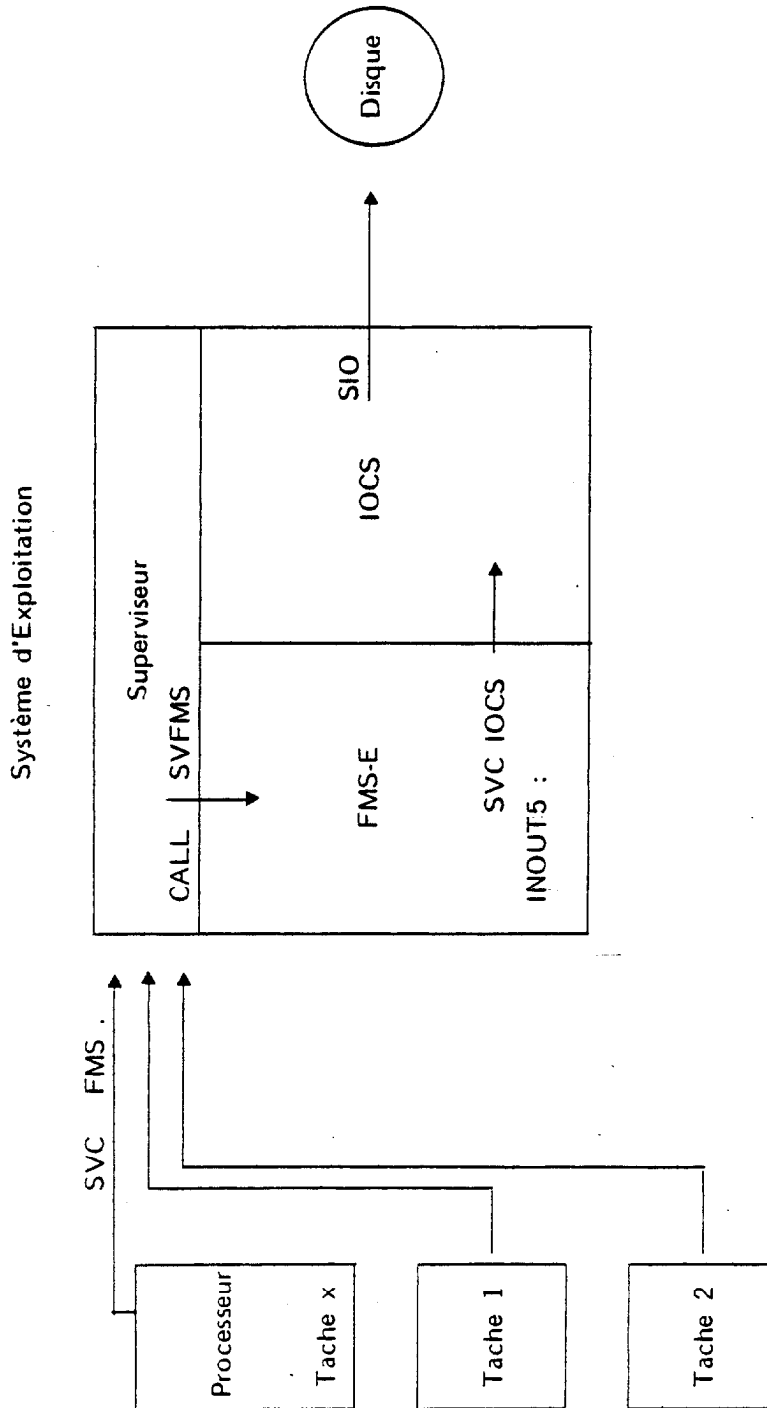
## 2 - STRUCTURE INTERNE

**Ce chapitre a pour but de décrire les différents interfaces internes utilisés par une méthode d'accès de FMS-E.**

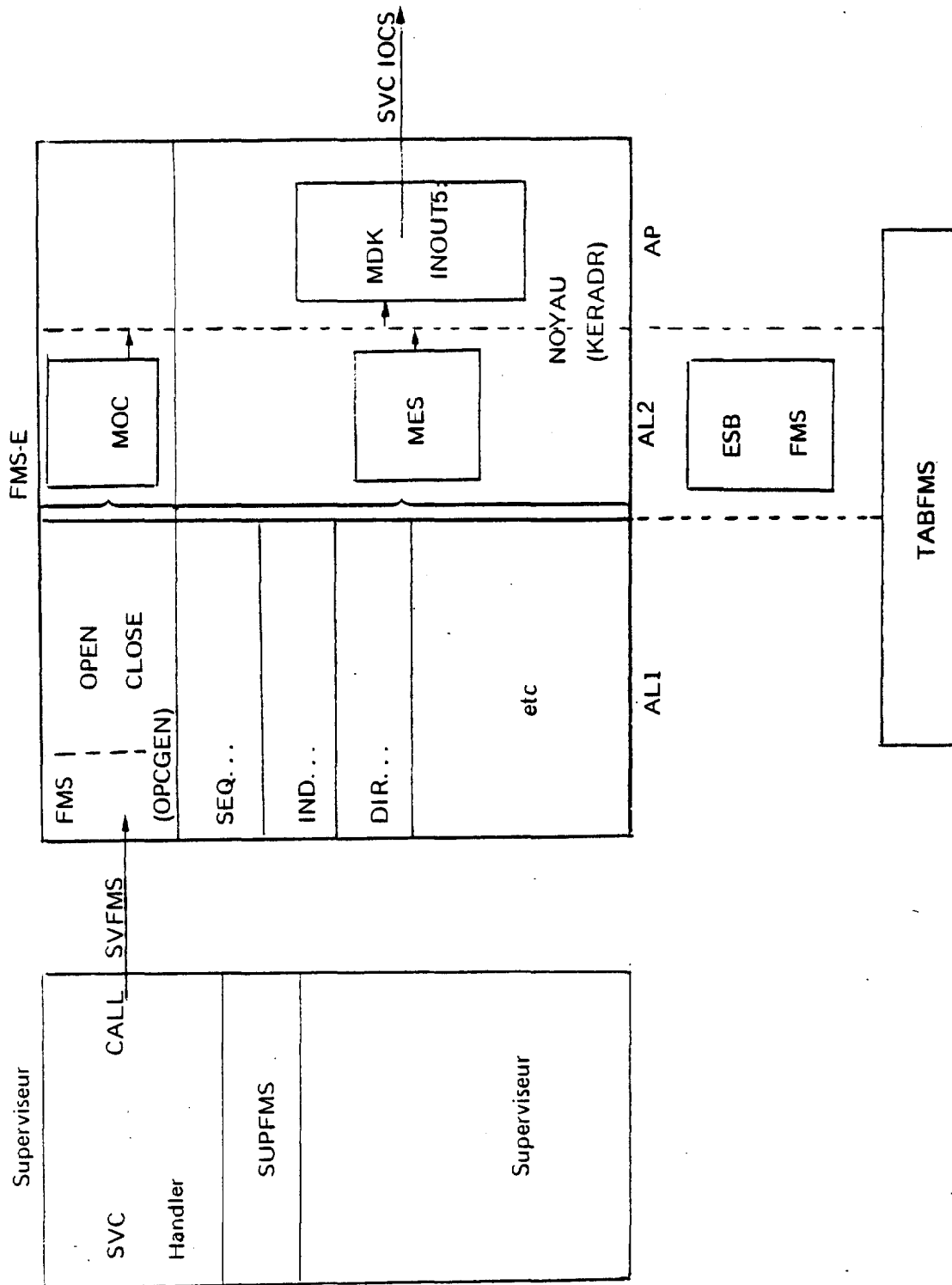
**Ces informations permettent une meilleure compréhension du fonctionnement interne de FMS-E, et permettent à un utilisateur ayant une bonne connaissance de la conception des systèmes de fichiers d'implémenter ses propres méthodes d'accès dans FMS-E.**

## 2.1 - PRESENTATION

### 2.1.1 - Schéma d'un Système d'Exploitation



2.1.2 - FMS-E et les éléments link-édités



### 2.1.3. - Description générale des modules

#### a) Le Superviseur

On distingue dans le Superviseur 2 parties qui fonctionnent en accord avec FMS-E.

- Le SVC Handler  
C'est la partie du superviseur qui transmet à FMS-E toutes les requêtes sur fichier.
- SUPFMS  
C'est un ensemble de sous-programmes réalisant des services Superviseur nécessaires au fonctionnement de FMS-E. (exemple : GETP).

#### b) Les modules de FMS-E

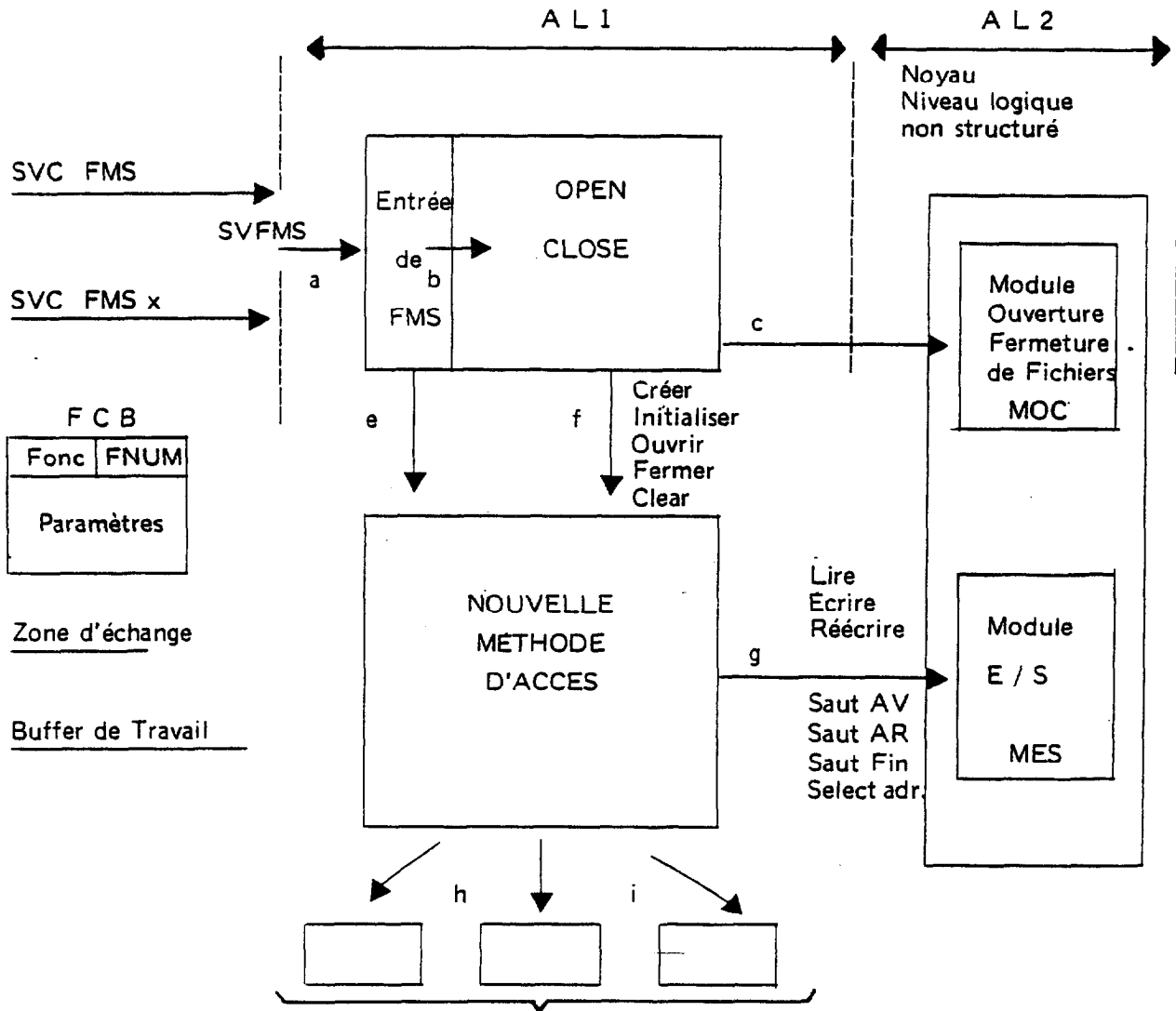
- Le module OPEN-CLOSE  
Exemples : OPCGEN, OPCPUB  
Ce module est divisé en 2 parties :
  - FMS : le tronc commun de traitement de toutes les requêtes fichier. Traitement de début et de fin commun à toutes les requêtes.
  - OPEN-CLOSE : La partie qui traite toutes les requêtes du niveau fichier. (ex : OPEN OLD).
- Le Noyau  
Exemples : KERADR  
Ce module est divisé en 2 anneaux ou 2 couches (voir schéma : Manuel de Présentation chapitre 1).
  - AL2 : anneau logique 2  
Cet anneau contient 2 sous-modules
    - MOC : qui gère les fichiers ouverts à un instant donné et qui permet de créer, détruire, les fichiers, etc...
    - MES : qui gère les E/S à un niveau logique non structuré. Le nième mot d'un fichier.
  - AP : anneau physique  
Cet anneau est principalement constitué d'un sous-module (MDK) qui gère dynamiquement un espace disque.
    - Création d'une chaîne de granules
    - Chargement en mémoire d'un DF
    - E/S physiques dans une chaîne de granules
    - etc...
- Le module de Bufferisation : ESBFMS  
ESBFMS est appelé par les méthodes d'accès. ESBFMS appelle le module MDK.  
C'est lui qui gère les lectures anticipées et les écritures dynamiques retardées.
- Le Séquentiel  
SEQBUF.  
Le Séquentiel gère une Unité d'Accès à
  - Les fichiers Séquentiel (n° 0)
  - L'accès à tout fichier en Séquentiel PUR
  - L'accès à un article en Portion d'Article Statique (PAS).
  - L'accès à un article en Portion d'Article Dynamique (PAD).
- L'Indexé  
INDBUF.  
L'indexé gère une Unité d'Accès à un fichier d'organisation Logique Indexé (n° 1).
- Le Direct  
DIRBUF.  
Le Direct gère une Unité d'Accès à un fichier d'organisation Logique Direct (n° 2).

- Le Direct à Trous  
Exemple : DITFMS  
Le Direct à Trous gère une Unité d'Accès à un fichier d'organisation Logique Direct (n° 2). C'est un sur-ensemble du Direct. Il comprend les 2 requêtes DCRE DSUP.
- Le Séquentiel Indexé  
Exemple : SIXFMS  
Le Séquentiel Indexé gère une Unité d'Accès à un fichier d'organisation Logique Séquentiel Indexé (n° 3).
- Le Séquentiel Chaîné  
Exemple : SCHFMS  
Le Séquentiel Chaîné gère une Unité d'Accès à un fichier d'organisation Logique Séquentiel Chaîné (n° 4).
- TABFMS  
Ce module contient principalement :
  - TDFU : la table des FU gérées par FMS-E
  - TABMA, TOPCMA : Tables d'aiguillage des 2 interfaces d'entrée des méthodes d'Accès.
- Le Direct Longueur Variable  
DIVFMS le Direct V gère une unité d'accès à un fichier d'organisation logique Direct V (n° 5).

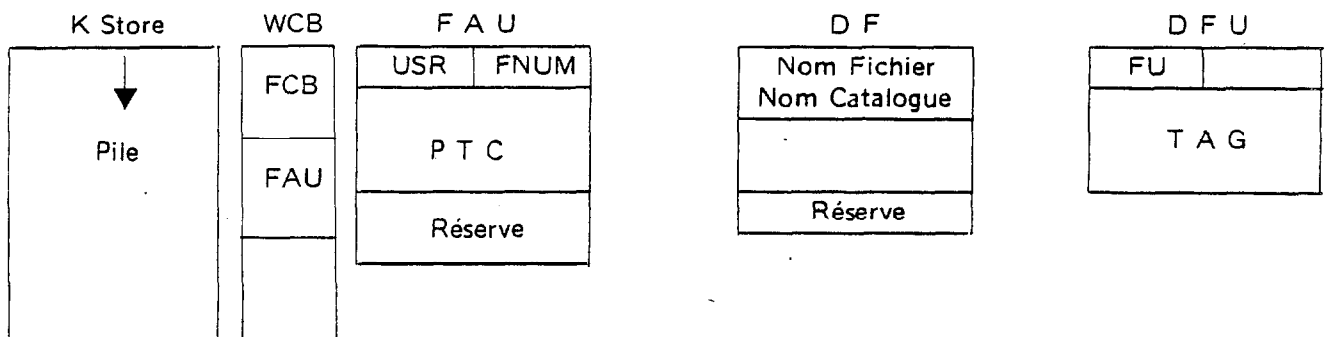


2.2. - DESCRIPTION GENERALE

2.2.1. - Environnement d'une Méthode d'Accès dans FMS-E



Sous-programmes de service FMS et Superviseur



## 2.2.2. - Présentation générale des interfaces

### a) Interface Superviseur / FMS

CALL SVFMS ;

Toutes les SVC envoyées par les différentes tâches sont prises en compte par le SVC Handler du Superviseur, qui les transmet à FMS-E par 1 seul point d'entrée : SVFMS.

Le tronc commun d'entrée dans FMS-E est situé dans le module OPEN-CLOSE ; on l'appellera : FMS.

Le label RETFMS est défini en externe pour une aide à la mise au point. Il se situe à la sortie de FMS.

### b) Interface FMS / OPEN-CLSE

CASE (RX) / NBOPC OF << RX := FONCT du FCB

FMS appelle l'OPEN-CLOSE, pour lui demander de traiter les SVC du niveau fichier.

Exemples au niveau fichier :

- SVC FMS pour un OPEN OLD
- SVC FMS pour un CREAT
- SVC FMS pour un RENAM
- SVC FMS pour un EOJ

### c) Interface OPEN-CLOSE / Noyau (MOC)

CALL MOC (RX = n° de primitive) ;

Le module OPEN-CLOSE appelle le Module Ouverture Fermeture (MOC) du Noyau pour lui demander de gérer l'entité fichier, le contenant fichier.

Exemples :

- Créer un fichier Temporaire ou Permanent
- Ouvrir un fichier Permanent
- Fermer un fichier Permanent
- Renommer un fichier
- Sauver un fichier Temporaire

Le label RETMOC est défini en externe pour une aide à la mise au point.

Il se situe à la sortie du module MOC du noyau.

### d) Interface Noyau (MDK) / IOCS

SVC (IOCS) ;

INOUT 5 :

Le module de gestion du disque (Granules) MDK appelle IOCS pour lui demander de réaliser les E/S physiques sur les FU :

- E/S d'informations système
- E/S d'informations utilisateur.

Le label INOUT5 : est défini en externe pour une aide à la mise au point. La SVC IOCS qui le précède traite tous les échanges disque demandés par FMS-E.

Pour les systèmes 64K : adresse de INOUT en mémoire = BEGIR-ENDVOL-1+@<sub>1</sub>+@<sub>2</sub>

où @<sub>1</sub> = déplacement de INOUT au link

@<sub>2</sub> = adresse d'implantation du système

Les labels IOCS34 et IOCS35 sont également définis en externe.

IOCS34 traite toutes les erreurs IOCS qui provoquent un '6034 FMS.

IOCS35 traite toutes les erreurs IOCS qui provoquent un '6035 FMS.

### e) Interface FMS / Méthode d'Accès

CALL TABMA (RX) ; << RX := n° de méthode d'accès

FMS appelle une Méthode d'Accès, pour lui demander d'exécuter les SVC d'accès au niveau article ou Portion d'Article.

- Exemples au niveau article :
  - SVC FMSD pour un DREAD
  - SVC FMSI pour un ISUP
- Exemple au niveau Portion d'Article (Séquentiel) :
  - SVC FMSS pour un READ

#### f) Interface OPEN-CLOSE / Méthode d'Accès

CALL TOPCMA (RX) ; << RX : = n° de méthode d'accès

Le module OPEN-CLOSE appelle une méthode d'accès lors de la réalisation des SVC suivantes du niveau fichier :

- SVC FMS pour - OPEN NEW
- OPEN OLD
  - CLOSE
  - CREAT
  - DELET
  - EOJ (USR)
  - EOJ (Généralisé)

Une méthode d'accès est appelée pour réaliser un complément de traitement spécifique de la méthode d'accès :

- Création de fichier
- Initialisation du contenu d'un fichier à sa création
- Ouverture d'un fichier
- Fermeture d'un fichier
- Droit d'accès pour un OPEN OLD

Le label RETMES est défini en externe pour une aide à la mise au point. Il se situe à la sortie du module d'entrée-sortie (MES) du noyau.

#### g) Interface Méthode d'Accès / Noyau (MES)

CALL MES (RX : = n° de primitive) ;

Une méthode d'accès appelle le Module d'Entrée-Sorties (MES) du Noyau, pour lui demander de gérer les accès au contenu d'un fichier.

Exemples :

- Lire les n mots suivants
- Sauter en avant de n mots
- Saut à la fin du fichier
- Sélectionner le mot d'adresse n (par rapport au début du fichier).

#### h) Interfaces FMS-E / SSP FMS

CALL spécifique

Le module OPEN-CLOSE contient des Sous-Programmes de service, utilisables par un quelconque module de FMS-E :

- VERFCB : Sous-programme vérifiant les adresses du FCB ou d'une table de codes d'arrêt.
- RADIX : Sous-programme de transformation de 8 caractères ASCII, en RADIX 40.
- VERBUF : Sous-programme. vérifiant les paramètres ABU, LBU définissant une zone mémoire utilisateur (FCB, Zone d'Echange, Buffer de Travail).  
En esclave par rapport à SLO SLE.  
En maître par rapport à SYSMEM.

Le Noyau contient 1 sous-programme de service utilisable par un quelconque module de FMS-E.

- PARBUF : Sous-programme de recherche d'un code d'arrêt dans une zone mémoire, dans le sens gauche-droite (Forward).

**i) Interface FMS-E / SUP FMS**

**CALL spécifique**

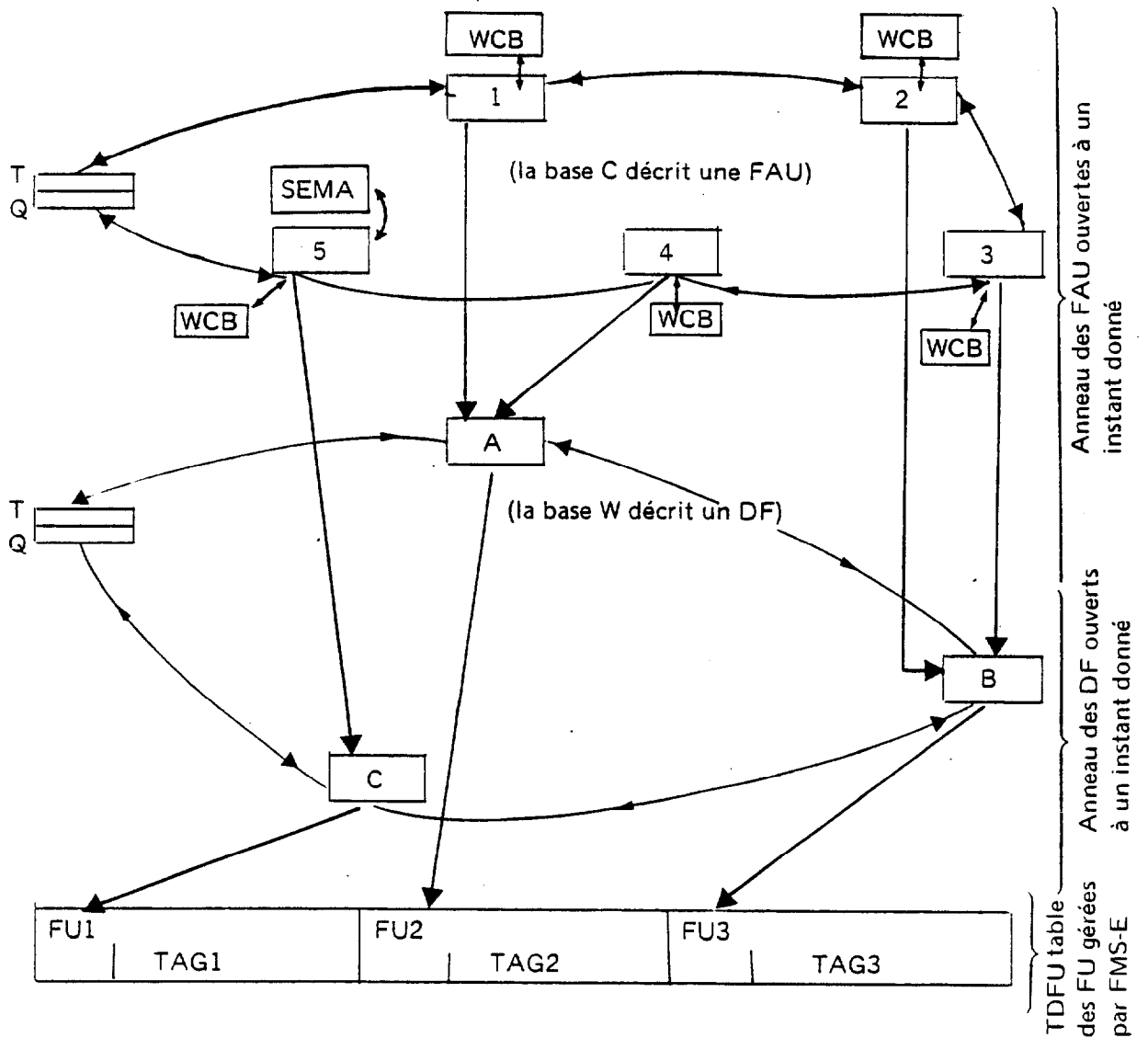
Une partie du Superviseur appelée SUPFMS contient 5 sous-programmes de service, utilisables par un quelconque module de FMS-E :

- ANASFU : Sous-programme de transformation d'une SU en FU avec contrôle de validité.
- GETP : Sous-programme d'allocation d'un pavé dans une des zones de pavés (ZUEP, ZIOCB, ZWCB, ZDF, ZBT8, ZBT9, ZBT10).
- FREEP : Sous-programme de désallocation d'un pavé.
- PVIN : Sous-programme de chaînage d'un pavé dans une liste-anneau. (FAU, DF, TLG).
- PVOU : Sous-programme de déchaînage d'un pavé.

### 2.2.3 - Les Tables du Système de fichiers

#### a) Architecture des tables

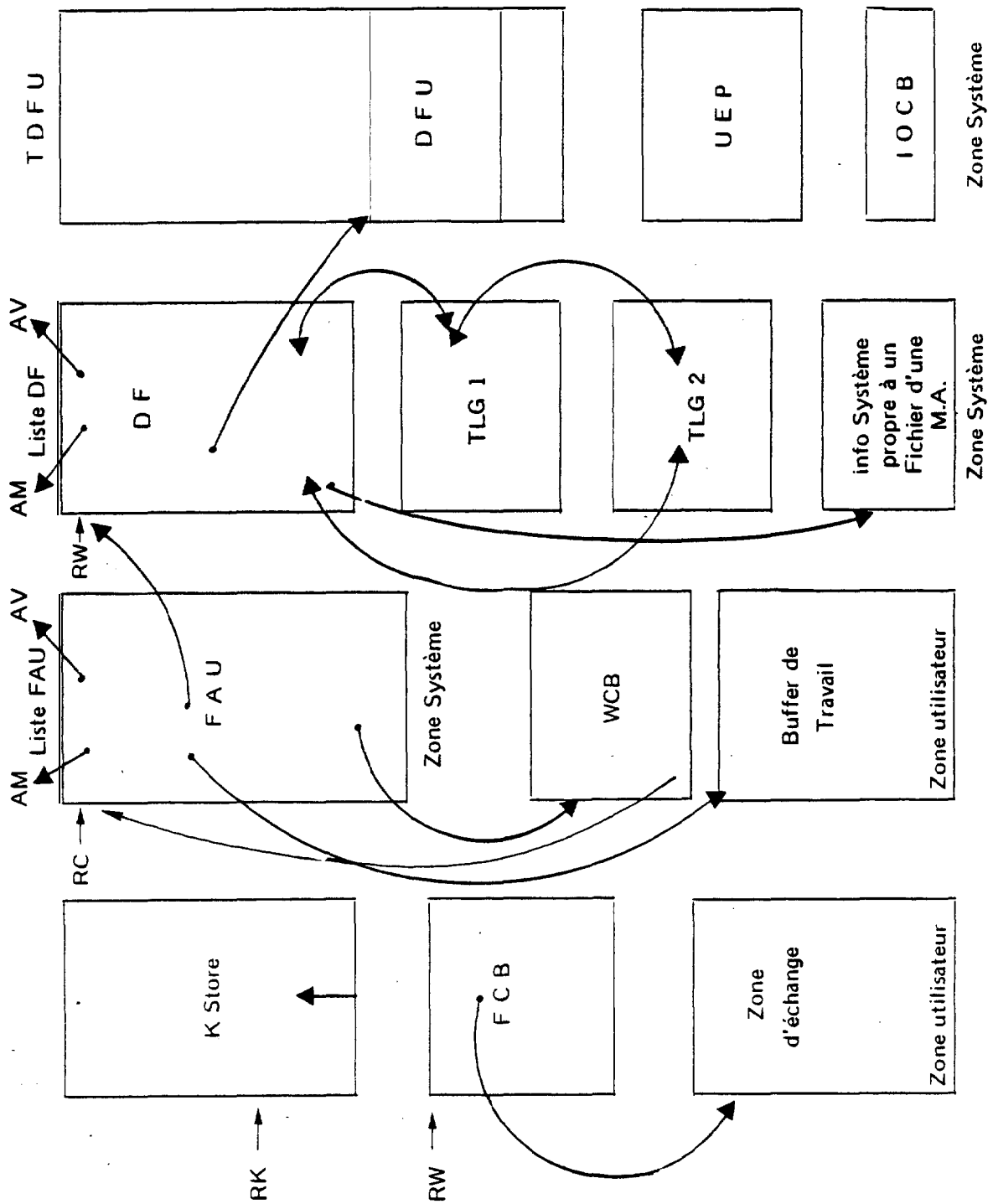
Exemple :



**Remarques :**

- Dans cet exemple l'anneau des FAU (unités d'accès à des fichiers ou chemins d'accès à des fichiers) contient 5 FAU caractérisés par l'argument USR-FNUM.  
Les 5 FAU étant : 1, 2, 3, 4, 5.
- L'anneau des DF comprend 3 DF, il y a donc 3 fichiers ouverts à l'instant donné respectivement
  - C ouvert sur FU1
  - A ouvert sur FU2
  - B ouvert sur FU3
- La FAU 5 accède au fichier C.
- Les FAU 1 et 4 accèdent au fichier A.
- Les FAU 2 et 3 accèdent au fichier B.
- Pour traiter chaque requête FMS alloue un WCB qu'il chaîne à la FAU correspondante pendant le traitement de la requête.
- La FAU 5 est une FAU publique ; il lui est associé un sémaphore de synchronisation des requêtes, dont le schéma est donné page 2-28.

b) Les zones de travail de FMS-E



c) La FAU : Le descripteur d'une Unité d'Accès à un Fichier

	0		7	8		15												
0	Pointeur		AVAL		AVALA													
1	Pointeur		AMONT		AMONTA													
2	USRA			FNUMA			USRFNU											
3	ASEMA / 0																	
4	ABUEP																	
5	LBUEP / 0																	
6	PBUEP																	
7				RW	LBUFSEC			LBUFSEC										
8	CS																	
9	CCC1																	
10	CCC																	
11	AM1																	
12	AM																	
13	AV1																	
14	AV																	
15		PC	MODMAI	BORN	SA	WERE	UTILSA	TIX	TIXMOD	EMA	TRTIX	NLG	TRBT	SID	S	W	RWK	PPFTYP
16	BOOLA																	
17	ADF																	
18	RESTA																	
19	DEJA																	
20	BORMAX																	
21	ATIX																	
22	LBTIX																	
23	PTIX																	
24	SLOTIX																	
25	SLETIX																	
26	SLOBUEP																	
27	SLEBUEP																	
28	AWCB																	
29																		



1°) FMSE

- chaînage de l'ensemble des FAU (2 mots) ouvertes à un instant donné

- 0 AVALA      pointeur sur la FAU suivante (pointeur aval)
- 1 AMONTA    pointeur sur la FAU précédente (pointeur amont)

- identification de la FAU (1 mot)

- 2 USRFNU    - USRA N° d'utilisateur
- FNUMA N° d'accès au fichier
- 3 ASEMA / o - Adresse du sémaphore pour une FAU publique (o sinon)

- buffers de travail de la FAU (3 mots)

- 4 ABUEP      Adresse du buffer en mémoire centrale (accès en mode maître) valide que si LBUEP = 0
- 5 LBUEP      longueur du buffer (en mots)  
              Par convention LBUEP = 0 ⇒  $\bar{A}$  buffer de travail
- 6 PBUEP      Pointeur dans le buffer de travail  
              Pointeur dans le buffer pour le code d'arrêt

- Booléens (1 octet droit)

- 7 LBUFSEC    8 booléens possibles

- le pointeur courant (7 mots)

- 8 CS            adresse du mot suivant dans le secteur courant
- 9 CCC1        = 0
- 10 CCC        adresse du secteur courant
- 11 AM1        = 0
- 12 AM        adresse du granule courant (adresse secteur)
- 13 AV1        = 0
- 14 AV        adresse du granule aval (suivant) (adresse secteur)

- Divers généraux (3 mots)

- 15 P P F T Y P - PP codage de la primitive précédemment exécutée sur cette FAU

    N° MA    ,    FONCT

Cas particulier du séquentiel : FONCT

'8X	=>	1	read	c. octets	
'AX	=>	3	write	c. octets	Pour l'Open Close
'CX	=>	5	read	c. arrêt	N° MA= 'F
'EX	=>	7	write	c. arrêt	
'7C	=>	C	skipb		
'7D	=>	D	skipf		
'7E	=>	E	rewind		
'7F	=>	F	skea		

- FTYP : - MAA N° de la méthode d'accès du fichiers ouvert pour cette  
4 bits FAU  
SEQ MAA = 0  
IND MAA = 1  
DIR MAA = 2

- S Simultanéité du fichier ouvert  
1 bit  
S = 0 Non simultané ou temporaire  
S = 1 Simultané

- W Protection écriture  
1 bit  
W = 0 écriture interdite  
W = 1 écriture autorisée

- RWK Codage de l'intervention  
2 bits lecture/écriture associée à l'OPEN  
de la FAU.

16 BOOLA 16 booleens de gestion de la FAU

- PC (0) primitive en cours sur la FAU si PC = 1
- MODMAI (1) mode de la tâche exécutant la requête à FMS  
(MODMAI = 1 mode maître)

Booleens géant la méthode d'accès séquentiel

- BORN (2) portion d'article en séquentiel pur (BORN = 1 => portion d'article)
- SA (3) sélection d'article  
(SA = 1 => Sélection d'article)
- WERE (4) écriture avec (allocation dynamique)  
réécriture (l'allocation est statique)  
(WERE = 0 => ECRI)  
(WERE = 1 => RECRI)

- UTILSA (5) la sélection d'article initialisée par une méthode d'accès (<sup>1</sup> SEQ) a été utilisée par le module d'accès séquentiel (UTILS A = 1).

Booleens gérant la méthode d'accès indexé

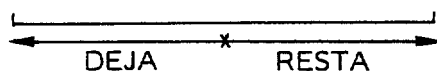
- TIX (6) (= 1) => la table d'index a été chargée en M.C. par IRTIX
- TIXMOD (7) (= 1) => la table d'index est modifiée par rapport à celle du disque (au niveau indexé par ISUP, IRNAM, INRITE).  
Ce bit n'a une signification que si TIX = 1
- MAJTIX (8) (= 1) => la table d'index est modifiée par rapport à celle du disque (au niveau Séquentiel portion d'article => MAJ de la taille du dernier article par RESTA et DEJA)
- NLC (10) Booleen qui commande si les écritures se font avec ou sans relecture de contrôle (0 = avec)

17 A D F Adresse du descripteur de fichier ouvert pour cette FAU

Divers particuliers aux méthodes d'accès (8 mots)

- Séquentiel (3 mots)

18 RESTA Définition du pointeur courant dans une portion d'article (en mots)  
19 DEJA déjà allouée



20 BORMAX Taille maximale en mots d'une portion d'article allocatrice.

- Indexé (5 mots) ou Indexé B

21 ATIX Adresse de la table d'index chargée en mémoire centrale ou de la portion de table d'index (1 secteur) chargée en MC

22 LBTIX Longueur de la table d'index chargée en mémoire centrale

24 SLOTIX SLO SLE associés à une table d'index  
25 SLETIX

26 SLOBUEP SLO SLE associés à un buffer de travail  
27 SLEBUEP

28 AWCB Adresse du WCB

- La bufférisation (module ESB)

- |   |         |   |
|---|---------|---|
| 4 | ABUF    | Adresse du buffer de bufférisation donné par l'utilisateur  |
| 5 | PFIN    | en lecture : pointe sur le mot suivant le dernier mot significatif du buffer.<br>en écriture : contient la longueur du buffer en mots   |
| 6 | PBUFF   | Pointe dans BUFF sur le prochain mot à lire ou à écrire   |
| 7 | LBUFSEC | - bit WR = 1 Bufférisation en écriture<br>(9) = 0 Bufférisation en lecture<br>longueur du buffer en nombre de secteurs d'ou longueur du buffer en octets LBUF = LBUFSEC X lg secteur LBUFSEC = 0 est caractéristique de "pas de bufférisation". |

- Booleens gégrant les FAU publiques

- |    |           |  |
|----|-----------|--|
| 16 | BOOLA     |  |
|    | TRTIX (9) | = 1 => une FAU publique nécessite pour son fonctionnement un buffer pour charger la TIX. Ce buffer sera alloué dans une des 3 zones 8, 9, 10 gérée par l'allocateur du superviseur. Ce buffer sera donc accessible par un ensemble de tâche. |
|    | TRBT (11) | = 1 => une FAU publique nécessite pour son fonctionnement un buffer de travail (OPEN NEW/OLD, CREATE). Ce buffer sera alloué et géré comme pour la TIX.  |

- Méthodes d'accès séquentiel chaîné

- |    |         |  |
|----|---------|--|
| 18 | AR1A    | ancienne valeur de AR1 en demi-secteur   |
| 19 | AR2A    | ancienne valeur de AR2 en demi-secteur   |
| 20 | PTBLOCA | pointeur sur disque du bloc dans le buffer de travail (en 1/2 secteur par rapport au début du fichier) |
| 21 | SBMAXA  | nombre maximal de sous blocs par bloc  |
| 22 | LSBA    | longueur du sous bloc  |
| 23 | NBSBA   | ancien nombre de sous bloc.  |

## 2°) FMSG

Autres définition de la FAU que celles du FMSE

- Le pointeur courant (7 mots)

8	CS	adresse du mot suivant dans le secteur courant
9	CCC1	poids fort de l'adresse du secteur courant
10	CCC	poids faible de l'adresse du secteur courant
11	AM1	poids fort de l'adresse du granule courant (adr. secteur)
12	AM	poids faible de l'adresse du granule courant (adr. secteur)
13	AV1	poids fort de l'adresse du granule aval (suivant) (adr. secteur)
14	AV	poids faible de l'adresse du granule aval (suivant) (adr. secteur).

d) Le WCB - Working Control Bloc

adr	nom	Module	But
0	WSLOBU	-	SLO de la zone d'échange (= 0 si zone maître)
1	WSLEBU	-	SLE de la zone d'échange (= 'FFF si WSLOBU = 0)
2	WFONFNU	-	N° requête, N° identificateur unité d'accès
3	WABU	-	Adresse relative à WSLOBU de la zone d'échange
4	WLBU	-	Longueur en octets de la zone d'échange
	WTCD	S	Adresse de la table de code d'arrêt
	WELBU	Si	Longueur en mots de la zone d'échange
	WELBUT	i	Longueur en mots de la zone d'échange
5	WPR	-	Compte-rendu
	WNCPLCOU	Si	Nombre de cases du poste courant
	WPRSC	SC	Zone de travail du séquentiel chaîné
6	WFNAM1	OPCL	Nom du fichier
	WACMOT	S	Événement fin d'échange
	WANUM	D,DT,DIV	Numéro d'article - Poids faible pour Direct à trou
	WANAM1	i	Nom d'article
	WAR1	SC	@ relative en 1/2 secteurs du 1er poste alloué à la chaîne
	WHOMRET	Si	Compte-rendu homonymie et écriture retardée
	WTANAME1	i	Nom radixé de l'article
	WANUMDT	DT	Mémoire conservant les poids faibles du ANUM utilisateur
	WABT	OPCL	Adresse relative dans buffer pour FMSMOVE
7	WFNAM2	OPCL	Nom du fichier
	WANAM2	i	Nom de l'article
	W M A X	Si	Nombre maximum d'articles dans un poste
	WAR2	SC	@ relative en 1/2 secteurs du dernier poste de la chaîne
	WADRS	DIV	Adresse relative en secteurs de l'article dans le fichier
	WTANAME2	i	Nom radixé de l'article
	WANUM1	DT	Poids fort du numéro de l'article
	WANUMDT1	DT	Mémoire conservant les poids forts du ANUM utilisateur
8	WFNAM3	OPCL	Nom du fichier
	WANAM3	i	Nom de l'article
	WMIN	Si	Nombre minimum d'articles dans un poste
	WORDR	SC	Pas de la lecture séquentielle
	WLONG	DIV	Longueur en octets d'un article
	WINECH	Si	Sauvegarde du compte-rendu INART pour une lecture
	WTANAME3	i	Nom radixé de l'article
9	WPUBW	OPCL	Nom du catalogue du fichier
	WANAM4	i	Nom de l'article
	WPAS	Si	Pas de la lecture séquentielle (SIRIS)
	WID1	SC	Identificateur binaire d'une chaîne
	WEXLO	DIV	Ancienne longueur sert à renuméroter
	WNCPLPREC	Si	Nombre de cases pleines dans le poste précédent le poste courant
	WLONGART	i	Longueur de l'article courant

adr	nom	Module	But
10	WFTYSFU WID2 WARALO WARSUIV WSAVPTX WPBU	OPCL SC Si Si i ESB	Numéro du FU et FTYP du fichier 2ième identificateur binaire d'une chaîne Numéro de rang du poste alloué Numéro de rang du poste suivant le poste courant Ancien PTIX sélectionné pour IRNAM Déplacement dans la zone d'échange
11	WLART WARP WEXNO WCLEL WARPREC WPBTIX WNBPA	OPCL SC DIV Si Si i SC	Longueur d'article Adresse du poste à sélectionner Ancien numéro d'article (sert à renuméroter) Longueur de l'enregistrement logique en octets Numéro du poste précédent le poste courant Déplacement dans le buffer table d'index
12	WTART WNB MOT WNPSAT	OPCL ESB Si	Taille d'un article ou poste Nombre de mots échangés Nombre de postes occupés
13	WNART  WLMOT WECR	OPCL  ESB Si	Nombre d'articles du fichier (nb de postes) Poids faible pour D.T. Longueur en mots du buffer Mémoire de travail de ACSYSA (en écriture)
14	WLCLE WLGMOVE  WABTIX WEXAD WNART1	OPCL Si  i DIV OPCL	Longueur de la clé Longueur du MOVE lors d'une réorganisation de postes Adresse de la portion de TIX Ancienne adresse (sert à renuméroter) Poids forts du nombre d'articles du fichier (Direct à Trous)
15	WABUEP WAZUEP WABUF	- DT ESB	Adresse du buffer de travail Adresse du pavé de la ZUEP donnée par le superviseur Adresse du buffer pour le buffering
16	WLBUEP WNARTDT1  WPFIN	- DT  ESB	Longueur du buffer pour le buffering Mémoire de travail contenant NART1 au départ et modifié ensuite Nombre de mots lus dans le buffer
17	WPBUEP WPBUFF  WNARTDT	Si, DIV ESB  DT	Pointe l'identificateur dans le buffer Pointe sur le prochain mot à lire ou à écrire dans le buffer Mémoire de travail contenant NART au départ et modifié ensuite

adr	Nom	Module	B u t																																			
18	WLBUFSEC	ESB	Longueur du buffer pour buffering																																			
19	WCS	NOY	Adresse mot suivant dans secteur courant																																			
20	WCCC1	NOY	Poids fort de l'adresse du secteur courant																																			
21	WCCC	NOY	Poids faible de l'adresse du secteur courant																																			
22	WAM1	"	Poids fort de l'adresse du granule courant (@ secteur)																																			
23	WAM	"	Poids faible de l'adresse du granule courant (@ secteur)																																			
24	WAV1	"	Poids fort de l'adresse du granule suivant (@ secteur)																																			
25	WAV	"	Poids faible de l'adresse du granule suivant (@ secteur)																																			
26	WPPFTYP	"	Codage primitive précédente et FTYP																																			
27	WBOOLA	-	Booleens de traitement d'une FAU																																			
			<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td> </tr> <tr> <td>PC</td><td>VODMAI</td><td>BORN</td><td>SA</td><td>NERE</td><td>JTILSA</td><td>TIX</td><td>TIXMOD</td><td>MAJTX</td><td>IRTX</td><td>NLC</td><td>TRET</td><td></td><td></td><td></td><td></td> </tr> </table>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	PC	VODMAI	BORN	SA	NERE	JTILSA	TIX	TIXMOD	MAJTX	IRTX	NLC	TRET							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																							
PC	VODMAI	BORN	SA	NERE	JTILSA	TIX	TIXMOD	MAJTX	IRTX	NLC	TRET																											
28	WEBOOLA	-	Booleens de traitement d'une requête																																			
			<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td> </tr> <tr> <td>RETAR</td><td>MAJHOM</td><td>HOMON</td><td>MAJART</td><td>L GART</td><td>MOFICH</td><td>RAZHOM</td><td>MAJNSLJ</td><td>TRANS</td><td>PARITE</td><td>HOMO</td><td>AFIK</td><td>COD1</td><td>SLEB1</td><td>COD2</td><td>SLEB2</td><td>NARW</td><td>ES</td><td>SAUT</td> </tr> </table>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	RETAR	MAJHOM	HOMON	MAJART	L GART	MOFICH	RAZHOM	MAJNSLJ	TRANS	PARITE	HOMO	AFIK	COD1	SLEB1	COD2	SLEB2	NARW	ES	SAUT
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15																							
RETAR	MAJHOM	HOMON	MAJART	L GART	MOFICH	RAZHOM	MAJNSLJ	TRANS	PARITE	HOMO	AFIK	COD1	SLEB1	COD2	SLEB2	NARW	ES	SAUT																				
29	WRESTA	S	Nombre de mots restant à parcourir jusqu'à la fin de la port. d'art. déjà allouée																																			
	WABUFC	Si	Adresse du buffer contenant le poste courant																																			
	WAR1A	SC	Ancienne valeur de AR1																																			
	WSAV1	Si	Contient LEL. Mémoire servant de communication entre CHAKST et INIMA																																			



adr	Nom	Module	But
30	WDEJA	S	Nombre de mots déjà parcourus depuis le début de la port. d'article
	WPBUFC	Si	Déplacement dans le buffer contenant le poste courant
	WAR2A	SC	Ancienne valeur de AR2
	WSAV2	Si	Contient LID. Mémoire servant de communication entre CHAKST et INIMA
31	WBORMAX	S	Taille maximum en mots d'une portion d'article allocatrice
	WRPCO	Si	Rang du poste courant de niveau 0
	WPTBLOCA	SC	Adresse disque du poste dans le buffer de travail
	WMOTSO	DT	Nombre de mots à sauter dans le secteur contenant le bit correspondant au ANUM
	WSAV3	Si	Contient TT. Mémoire servant de communication entre CHAKST et INIMA
32	WATIX	i	Adresse du buffer contenant la table d'index
	WRPC1	Si	Rang du poste courant de niveau 1
	WABLOC	DIV	Adresse du bloc de TIX contenu en mémoire
	WSAV4	Si	Contient NEGE. Mémoire servant de communication entre CHAKST et INIMA
	WSBMAXA	SC	Nombre maximal d'articles par poste
33	WLBTIX	i	Longueur du buffer de table d'index
	WRPC2	Si	Rang du poste courant de niveau 2
	WSAV5	Si	Contient NEPS. Mémoire servant de communication entre CHAKST et INIMA
	WLBUDT	DT	Longueur du buffer utilisateur
	WLSBA	SC	Longueur du sous-bloc
34	WPTIX	i, DIV	Pointeur courant dans la table d'index
	WRPC3	Si	Rang du poste courant de niveau 3
	WNBSBA	SC	Ancien nombre de sous bloc
35	WSLOTIX	i	Valeur du SLO associé au buffer de la table d'index
	WNCOU	Si	Numéro du niveau courant
	WSESO1	DIT	Nombre de secteurs à sauter. Poids fort
	WTLART	DIV	Taille de l'article
36	WSLETIX	i	Valeur du SLE associé au buffer de la table d'index
	WARCOU	Si	Numéro de rang du poste courant
	WNUM	DIV	Numéro de l'article de la requête précédente
	WSESO	DT	Nombre de secteurs à sauter. Poids faible
	WLSEC2	SC	Taille d'un demi secteur

adr	Nom	Module	But
37	WSLOBT	NOY	Valeur du SLO associé au buffer de travail
38	WSLEBT	"	Valeur du SLE associé au buffer de travail
39	WLAM1	"	Sauvegarde poids fort de l'adresse du granule amont
	WQT2	"	Mémoire de travail
	WAORG	-	Adresse origine d'un MOVE 1024K
	WPBUF	NOY	Mémoire de travail de SELECH
40	WLAM	NOT	Sauvegarde poids faible de l'adresse du granule amont
	WDIV	-	Contient le diviseur pour DIVISE
	WMUL	-	Contient le multiplicateur pour MULTIP
41	WLAV	NOY	Sauvegarde poids fort de l'adresse du granule aval
	WQT1	-	Mémoire de travail de MULTIP et de DIVISE
42	WLAV1	NOY	Poids faible de l'adresse du granule aval
	WADEST	-	Adresse destination d'un MOVE 1024K
	WRS1	-	Mémoire de travail de MULTIP et de DIVISE
43	WADR1	NOY	Nombre de mots pour sélection (Poids fort)
	WMESCI	"	Nombre de mots à échanger pour MES
	WLG MV	-	Longueur d'un MOVE 1024K
44	WADR2	NOY	Nombre de mots pour sélection (Poids faible)
	WATCD	"	Flag de test pour le code d'arrêt
	WFUFTYP	"	FTYP pour MOC
45	WAZU	"	Adresse de la zone d'échange pour MES
	WAZE	"	Adresse de la zone d'échange pour MDK
	WASEMA	OPCL	Adresse du sémaphore si FAU publique
46	WSLOZE	NOY	Valeur du SLO associé à la zone d'échange
47	WSLEZE	"	Valeur du SLE associé à la zone d'échange
48	WSLOMDK	"	Valeur du SLO de la zone d'échange pour MDK
	WSLOORG	-	Mémoire de travail de la procédure MVMIL (MOVE 1024K)
49	WMDKCO	NOY	Compte de mots échangés pour MDK
	WMESCO	"	Compte de mots échangés pour MES
	WNGR	"	Compteur de granules (CRET LG - SELECH)

adr	Nom	Module	But
50	WMDKC1 WSLEORG	NOY -	Compte de mots à échanger pour MDK (poids faible) Mémoire de travail de la procédure MVMIL (MOVE 1024K)
51	WMDKC2 WSLODEST	NOY -	Compte de mots à échanger pour MDK (Poids fort) Mémoire de travail de la procédure MVMIL (MOVE 1024K)
	WSAUV1	NOY	Mémoire de sauvegarde pour le noyau
52	WATLG WSAUV WPTAB WBUFTIX	NOY " i i	Pointeur sur la TLG dans SELECH Mémoire de sauvegarde pour le noyau
	WSLEDEST	-	Pointeur sur la table d'index (INIMA). Remise à zéro de la TIX Mémoire de travail de la procédure MVMIL (MOVE 1024K)
53	WSAVAR1	i	Poids fort de l'@ relative dans le fichier de l'article sélectionné
	WANCNUM1	DT	Mémoire conservant les poids forts du ANUM utilisateur (ANUM1)
	WLTG	NOY	Mémoire de travail du noyau (CRETG)
54	WSAVAR	i	Poids faible de l'@) relative dans le fichier de l'article sélectionné
	WANCNUM	DT	Mémoire conservant les poids faibles du ANUM utilisateur (ANUM)
	WBCLEF	NOY	Mémoire de travail du noyau (CRETG - CHATLG)
55	WAD1 WLDF	" "	Sauvegarde des poids forts de l'adresse disque Sauvegarde de la longueur du DF
56	WAD WADEG	" "	Sauvegarde des poids faibles de l'adresse disque Sauvegarde adresse début espace granule
57	WLGMO	"	Mémoire de travail du noyau - ENTSOR - nombre de mots
	WCOUR1 WADTF	" "	Sauvegarde poids fort @ granule (GAV -GAR) Sauvegarde de l'adresse de début de la table des fichiers
58	WMOECH WCOUR WATF	" " "	Nombre de mots échangés ENTSOR Sauvegarde poids faible @ granule (GAV - GAR) Sauvegarde de l'adresse de fin de la table des fichiers

adr	Nom	Module	But
59	WFPAV WLGMAX	NOY "	@ de fin d'un pavé de TLG (GAV - GAR) Longueur Max. en mots d'une TLG (MOVTLG)
60	WPRE1 WLGLIG	" "	Sauvegarde poids forts @ granule (GAV - GAR) Longueur d'une ligature (1 ou 2 mots)
61	WPRE WESMAX	" "	Sauvegarde poids faible @ granule (GAV - GAR) Longueur d'1 E/S Max. selon type de disque (ENTSOR)
62	WSLO	-	SLO courant
63	WSLE	-	SLE courant
64	WFAU	-	Adresse du pavé FAU.

Le WCB résumé

0	WSLOBU	}	Recopie du FCB
1	WSLEBU		
2	WFONFNU		
3	WABU		
4	WLBU		
5	WPR		
6	WFNAM1		
7	WFNAM2		
8	WFNAM3		
9	WPUBW		
10	WFTYSFU		
11	WLART		
12	WTART		
13	WNART		
14	WLCLE		
15	WABUEP	}	Recopie de la FAU
16	WLBUEP		
17	WPBUEP		
18	WLBUFSEC		
19	WCS		
20	WCCC1		
21	WCCC		
22	WAM1		
23	WAM		
24	WAV1		
25	WAV		
26	WPPFTYP		
27	WBOOLA		
28	WEBOOLA	}	Recopie de la FAU
29	WRESTA		
30	WDEJA		
31	WBORMAX		
32	WATIX		
33	WLBTIX		
34	WPTIX		
35	WSLOTIX		
36	WSLETIX		
37	WSLOBT		
38	WSLEBT		

39	WLAM1	}	Mémoires de travail
40	WLAM		
41	WLAV		
42	WLAV1		
43	WADR1		
44	WADR2		
45	WAZU		
46	WSLOZE		
47	WSLEZE		
48	WSLOMDK		
49	WMDKCO		
50	WMDKCI		
51	WMDKC2		
52	WATLG		
53	WSAVAR1		
54	WSAVAR		
55	WAD1		
56	WAD		
57	WLGMO		
58	WMOECH		
59	WFPV		
60	WPRE1		
61	WPRE		
62	WSLO		
63	WSLE		
64	WAFU		

e) Sémaphore de FAU publique (voir manuel de Référence 3.5.3)

	0	7 8	15
0	@FAU		
1	USRP		U S R
2	CPTRA		
3	USRC		USRPRE
4	NOTACH		
5	SEMFAU		
6	File de bits de SEMFAU		
7			
8			
9			
10			
11			
12	N° ZBT pour BT		
13			
14	N° ZBT pour TIX (Indexé)		
15			
24			

f) Le DF - Descripteur de fichier

	0	7	8	15					
0	Pointeur AVAL - AVALF								
1	Pointeur AMONT - AMONTF								
2	FNAM1								
3	FNAM2								
4	PUBW								
5	LTLGD	EMA	SID	S	W	AFI	OFI	LTGFTYP	
6	FS								
7	FF1								
8	FF								
9	DD1								
10	DD								
11	TART'								
12	NART1'								
13	NART'								
14	U	S	R	F	CPTFI	USRCPI			
15	ADKDF								
16	ADFU								
17	ATLG								
18	AFTLG								
19	LTLG	RWKF	MORC		AFID16	IDENT	VIDE	MODF	BOOLF
20	LTG								
21	LSEC								
22	NZUEP	LDF							
23	ADRSF								
24	LTIX / APS / NOSUP								
25	ABLIBF								
26	NBLIBF								
27	ADTF								
28	AFTF								
29	ADEG								



## Description d'un DF

- Zone associée à un fichier ouvert (temporaire ou permanent) (zone en M.C.).

N° mot - Chaînage de l'ensemble des DF (2 mots)  
des fichiers ouverts à un instant donné

0 AVALF Pointeur sur le DF suivant (pointeur aval)  
1 AMONTF Pointeur sur le DF précédent (pointeur amont)

- Caractéristiques du fichier (4 mots)

2 FNAM1 } 6 caractères imprimables codés en RADIX 40  
3 FNAM2 }  
4 PUBW "mot de passe public" ou nom du catalogue  
2 caractères imprimables codés en RADIX 40 et cadrés à droite

TP bit 1 de PUBW

= 0 fichier permanent  
= 1 fichier temporaire

5 LTGFTYP - LTLGD  
- FTYPT (octet droit)

EMA  
(bits 9/10)

SID  
(bits 11/12) = 0000 fichier séquentiel  
= 0001 fichier indexé  
= 0010 fichier direct  
= 0011 fichier séquentiel indexé  
= 0100 fichier séquentiel chaîné

SF = 1 fichier simultané  
(bit 13) = 0 fichier non simultané

WF = 0 protection écriture  
(bit 14) = 1 non protection écriture

AFI = 1 fichier en accès direct physique  
(bit 15) = 0 fichier en accès séquentiel physique

OFI = 1 fichier à organisation directe physique  
(bit 16) = 0 fichier à organisation séquentielle physique

## - Pointeurs (5 mots)

6	FS	Position dans le secteur du mot suivant le dernier mot du fichier
7	FF1	= 0
8	FF	Adresse disque du dernier secteur du fichier
9	DD1	= 0
10	DD	Adresse disque (en secteur) du premier granule du fichier

## - Précisions du niveau article (2 mots)

11	TART'	Taille des articles si le fichier est direct
12	NART1'	= 0
13	NART'	Nombre d'articles si fichier direct Nombre maximal d'articles si fichier indexé

## - Diverses variables (niveau disque 6 mots)

14	USRCPI	- USRF N° de l'utilisateur courant - CPTFI Compteur d'utilisation du fichier (nb de FAU sur ce fichier)
15	ADKDF	@ du secteur sur le disque qui contient le descripteur du fichier (seulement si fichier permanent, sinon valeur quelconque)
16	ADFU	@ mémoire de la table DFU décrivant la FU
17	ATLG	@ mémoire du premier pavé contenant la TLG (TLG = table des ligatures de granule)
18	AFTLG	@ mémoire du dernier pavé contenant la TLG
19	LTLGF	- LTLG longueur en mots de la TLG - BOOLF

RWK	}	=	0	- mono accès
(bits 9/10)				
WWF				
RRF	3	- nb quelconque en lecture et nb quelconque écriture		

	MOFIC	= 1	le contenu du fichier a été modifié
	IDENT	= 1	fichier trouvé dans une recherche dans FIFI
	VIDE	= 1	- IDENT= 1 .1 place libre dans même secteur que DF trouvé - IDENT= 0 .1 place libre dans dernier secteur de FIFI
	MODF	= 1	le MCDF doit être mis à jour sur disque
20	LTG		longueur du granule (en secteurs)
21	LSEC		longueur du secteur en mots
22	NZUEP		(octet gauche) numéro de zone de l'allocateur, contenant des pavés de la dimension voulue

## - Méthode d'accès indexé

24	LTIX	longueur de la table d'index (en mots)
----	------	--

- Méthode d'accès séquentiel chaîné

11	TART'	taille du bloc en mots £ 6K mots
12	NART1'	= 0
13	NART'	nombre de blocs du fichier
25	ABLIBF	adresse du premier bloc libre, en 1/2 secteur par rapport au début du fichier
26	NBLIBF	nombre de blocs libres (NART au create ou à l'OPEN NEW)

- Méthode d'accès séquentiel indexé

24	APS	adresse de l'extension du DF
----	-----	------------------------------

- Méthode d'accès direct longueur variable

23	ADRSF	adresse de fin du fichier (adresse secteur)
24	NOSUP	le plus grand numéro d'article

3°) FMS-G

Autres définitions du DF que celles du FMS-E

- Caractéristiques du fichier

5	LTLGD	(octet gauche) nombre de granules du fichier
---	-------	--

- Pointeurs

7	FF1	Poids fort de l'adresse disque du dernier secteur du fichier
8	FF	Poids faible " " " " " " " "
9	DD1	Poids fort de l'adresse disque du premier granule
10	DD	Poids faible " " " " " " " "
12	NART1'	Poids fort du nombre d'article
13	NART'	Poids faible du nombre d'article

- Diverses variables

19	BOOLF	
	AFID16	- sauvegarde du bit AFI de LTGFYTP

- Variables issues du type de FU

22	LDF	(octet droit) longueur du DF disque en mot = 8 si petite FU < 4 Mega mots = 12 si grande FU > 4 Mega mots
27	ADTF	adresse de début de la table des fichiers (adresse secteur)
28	AFTF	adresse de fin de la table des fichiers (adresse secteur)
29	ADEG	adresse de début de l'espace granule (adresse secteur).

**Méthode d'accès séquentiel indexé**

**- Extension du DF : PS (Poste Système)**

**N° mot**

0	LID	longueur identificateur en-mots
1	LEL	longueur élément logique en mots
2	NEPS	nombre d'éléments par poste supérieur
3	NEGE	nombre d'éléments par poste niveau 0
4	NIV	octet contenant le no du plus haut niveau
	TT	2 bits indiquant le type de table 00 table directe avec homonymes 01 table directe sans homonymes 11 table inverse
5	RPS	rang du poste de plus haut niveau
6	NAL	nombre d'articles (ou postes) libres
7	RAVC	rang du premier poste libre chaîné
8	RALS	rang du premier poste libre en séquence (à la fin du fichier)
9	WTART	sauvegarde de la taille du poste
10	ADRDF	adresse du DF (MCDF)

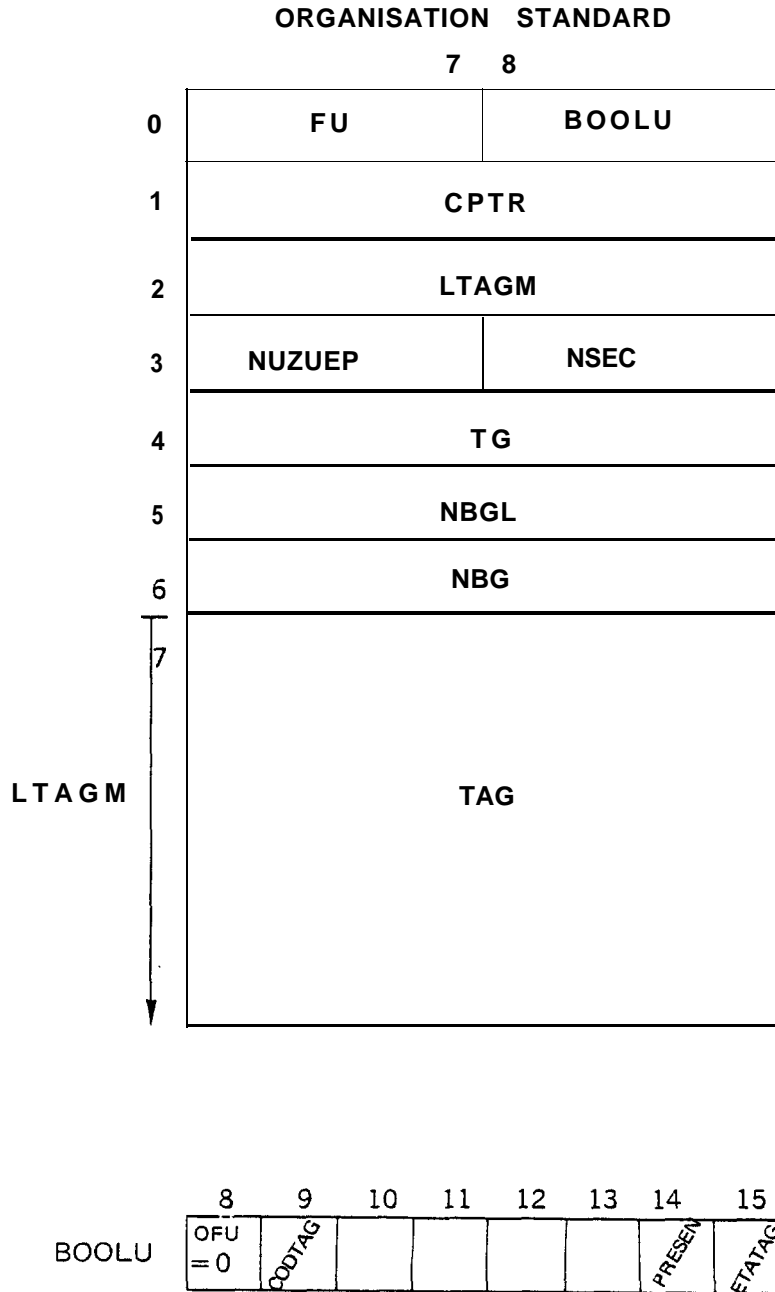
- Extension du DF pour le Séquentiel Indexé

Extension DF		
0	LID	mots
1	LEL	
2	NEPS	
3	NEGE	
4	NIV	TT
5	RPS	
6	NAL	
7	RAVC	
8	RALS	
9	WTART	
10	ADRDF	

TT = 00    Table directe avec homonymes  
       = 01    Table directe sans homonymes  
       = 11    Table inverse

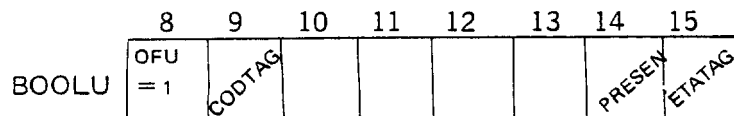
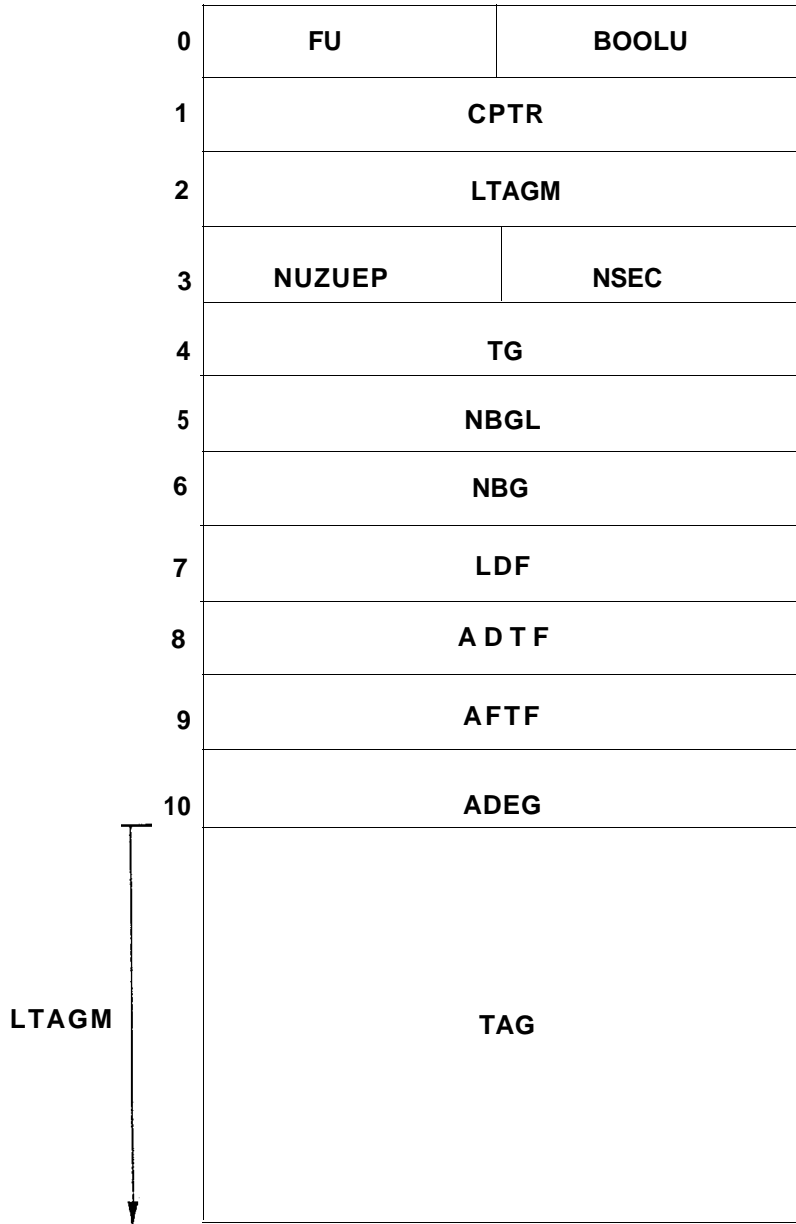
g) Le DFU : Le descripteur de FU (unité fonctionnelle)

1°) DFU 4 Mega :



2°) DFU 50 Mega

ORGANISATION GRAND DISQUE  
7 8





1°) FU 4 MEGA

0	FU	N° de FU défini à la configuration (GENFMS-E)
	BOOLU	Booleens de gestion DFU + TAG
	CODTAG=0	LTAGM est codé sur l'octet gauche
	=1	LTAGM est codé sur un mot
	PRESEN=1	Indique la présence de la TAG en MC
	ETATAG=1	Indique une modification de la TAG.
1	CPTR	Compteur de fichiers ouverts sur la FU (GENFMS-E)
2	LTAGM	Longueur de la TAG maxi gérée par cette FU (GENFMS-E) nb de mots de la chaîne de bits
3	NUZUEP	Numéro de la zone de pavé contenant les secteurs (UEP) de la taille correspondante
	NSEC	tel que taille du secteur = 2 <sup>NSEC</sup>

Informations chargées à partir du disque

4	TG	Taille du granule sur le support (en secteurs) initialise par FUP4
5	NBGL	Nombre de granules libres sur le support initialisé par FUP4
6	NBG	Nombre de granules maxi sur le support initialisé par FUP4
7-127	TAG	Table d'allocation des Granules du support initialise par FUP4. MAJ par l'allocateur de granules.

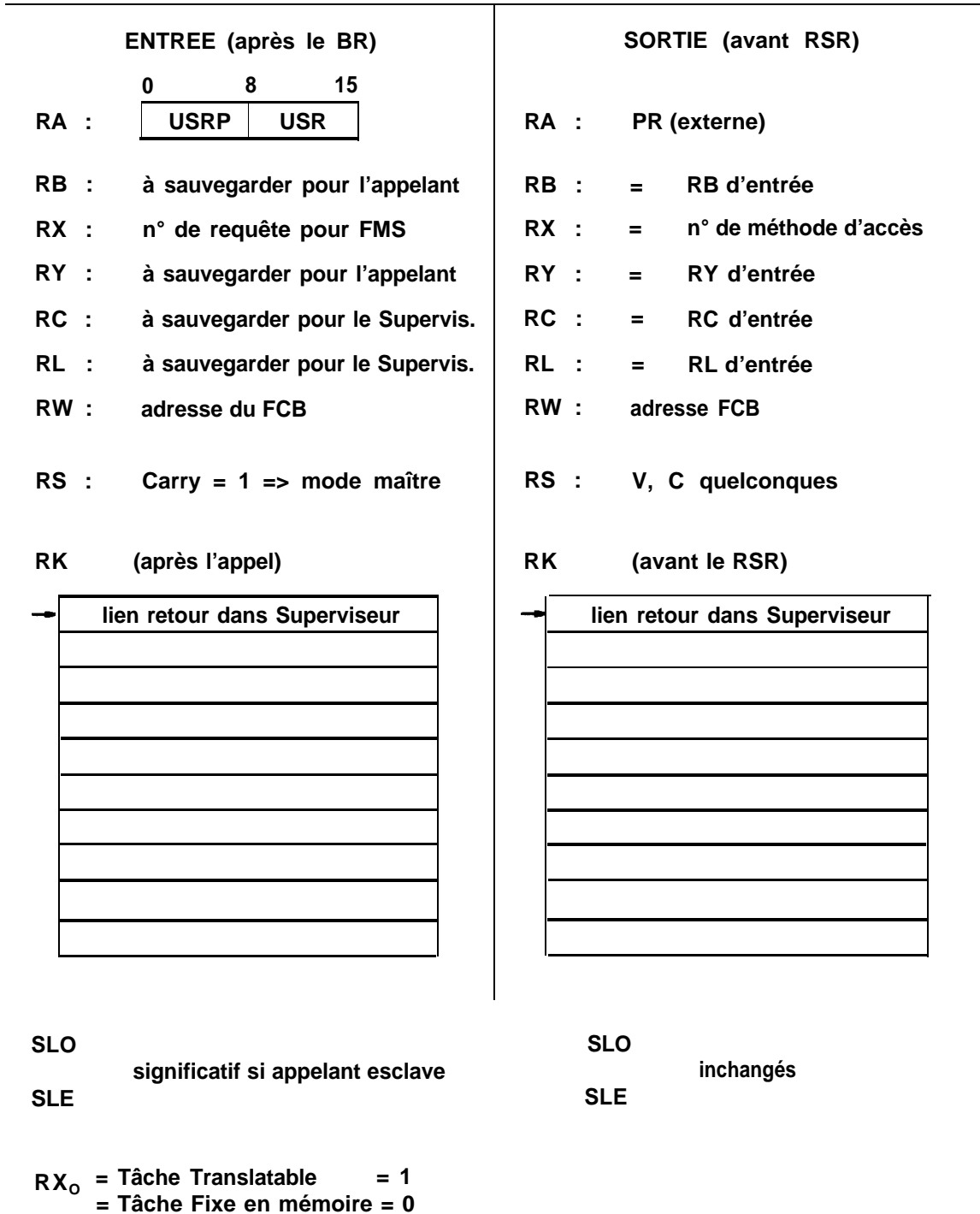
2°) FU 50 MEGA

Les mots 0 à 6 ont la même signification que pour une FU 4 Mega

7	LDF	Longueur du DF, calculée par FUP4 en fonction de la taille de la FU = 8 si taille FU < 4 Mega = 12 si taille FU > 4 Mega
8	ADTF	Adresse de début de la table des fichiers, calculée par FUP4 (adresse secteur)
9	AFTF	Adresse de fin de la table des fichiers, calculée par FUP4 (adresse secteur)
10	ADEG	Adresse de début de l'espace granule, calculée par FUP4 (adresse secteur)
11-2047	TAG	Table d'Allocation des granules du support initialisé par FUP4. MAJ par l'allocateur de granules.

2.2.4. - Interface : Superviseur / FMS

Point d'entrée dans FMS : Label = SVFMS



## a) Paramètres d'entrée de l'interface : SUPERVISEUR / FMS

- \* L'octet droit du registre RA contient le numéro de l'utilisateur.  
L'octet gauche contient le numéro d'utilisateur publique.

USRP	USR	0	£	USR	£	255
		0	£	USRP	£	255

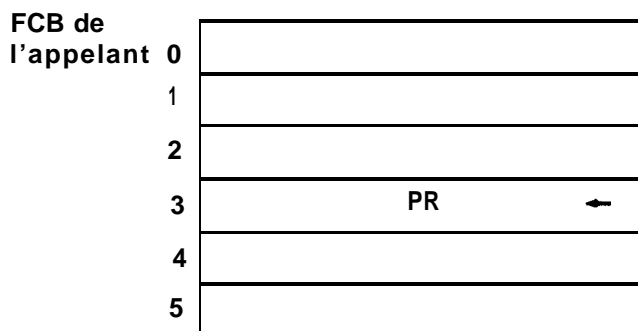
- \* Le registre RB sera sauvegardé et restauré par FMS-E
- \* Le registre RX contient le numéro de requête pour FMS-E  
Le bit 0 de RX indique si la tâche esclave est translatable en mémoire (DRPS).

FONCTION	SVC	Numéro de Requête	Numéro de Méth. Accès
Niveau Open Close	FMS	'38	- 1
Séquentiel	FMSS	'39	0
Indexé	FMSI	'3A	1
Direct et direct à Trous	FMSD	'3B	2
Séquentiel Indexé	FMSX	'28	3
Séquentiel Chaîné	FMSC	'29	4
Direct Longueur Variable	FMSV	'2A	5
		'2B	6
		'2C	7
		'2D	8

- \* Le registre RY sera sauvegardé et restauré par FMS-E
- \* Le registre RC sera sauvegardé et restauré par FMS-E (base RC du SVC Handler)
- \* Le registre RL sera sauvegardé et restauré par FMS-E (base RL du SVC Handler)
- \* Le registre RW contient l'adresse du FCB associé à la requête de l'appelant.  
Remarque : si l'appelant est en mode esclave c'est une adresse par rapport à SLO
- \* Dans le registre RS le carry précise le mode de l'appelant.
  - carry = 0 => appelant en mode esclave
  - carry = 1 => appelant en mode maître
- \* Le registre RK pointe sur la KSTORE du 1er appelant (bit SVCS) (au moins 60 mots de pile correctement utilisables).

b) Paramètres de sortie de l'interface : SUPERVISEUR / FMS

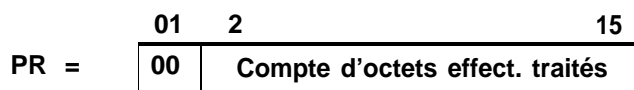
- \* Le registre RA contient le paramètre de retour chargé dans le FCB de l'appelant (4e mot du FCB)



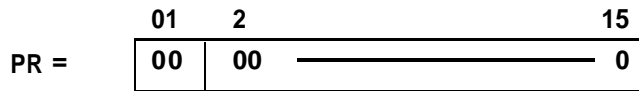
- \* PR : Paramètre de retour

1°) Requête terminée normalement

a) Requêtes sur portion d'article (SVC FMSS)



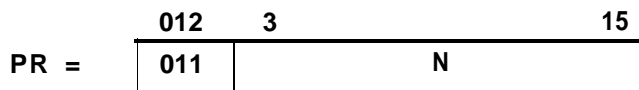
b) Autres requêtes



2°) Requête terminée anormalement pour une cause Hardware



3°) Requête terminée anormalement pour une cause Logique



N représente le codage de l'erreur logique.

- \* Le registre RB est inchangé
- \* RX contient le numéro de Méthode d'Accès
- \* Le registre RY est inchangé
- \* Le registre RC est inchangé
- \* Le registre RL est inchangé
- \* Le registre RW contient l'adresse du FCB associé à la requête de l'appelant.
  
- \* Le registre RS est inchangé, par contre les indicateurs V et C (overflow et carry) sont quelconques.
- \* La pile K de l'appelant pointée par RK se trouve dans le même état qu'à l'entrée dans FMS-E.

Remarques - les registres SLO et SLE sont inchangés.  
- les mémoires définissant la CDA ('18, '19) sont inchangés.

#### **2.2.5. - Interface FMS-E / SUPFMS**

- Cet interface permet à FMS-E en particulier aux Méthodes d'Accès d'accéder à des informations gérées par le superviseur et non fournies par l'usager.

Accès à :

FU : Unité fonctionnelle valide à laquelle a été affectée une Unité Symbolique.  
Sous-programme ANASFU

L'allocateur dynamique de pavés en mémoire centrale. Sous-programmes : GETP, FREEP, PVIN, PVOU.

a) Interface : FMS-E (OPEN-CLOSE) / ANASFU

Appel : CALL ANASFU ;

ENTREE (après le CALL)	SORTIE (avant R S R)																						
<p>RA      "             SU/FU       </p>	<p>RA : - si OK (SU affectée à FU ou n° de FU valide) :</p> <p style="padding-left: 40px;">       '00             F U       </p> <p>- sinon :</p> <p style="padding-left: 40px;">       non nul             "       </p>																						
<p>RB :      "</p>	<p>RE : détruit</p>																						
<p>RX :      "</p>	<p>RX : détruit</p>																						
<p>RY :      "</p>	<p>RY : détruit</p>																						
<p>RC :      "</p>	<p>RC : inchangé</p>																						
<p>RL :      "</p>	<p>RL : inchangé</p>																						
<p>RW :      "</p>	<p>RW : inchangé</p>																						
<p>RS :      "</p>	<p>RS : inchangé</p>																						
<p>RK      KSTORE de l'appelant</p>	<p>R K      inchangé</p>																						
<p>→ <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td style="text-align: center;">lien appelant</td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> </table></p>	lien appelant											<p>→ <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td style="text-align: center;">lien appelant</td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> <tr><td> </td></tr> </table></p>	lien appelant										
lien appelant																							
lien appelant																							

## b) Généralités sur l'allocateur de Pavés

Allocation dynamique de pavés en mémoire centrale. FMS-E a besoin de pavés en mémoire centrale pour gérer plusieurs types d'information.

- 1°) - Les UEP : L'Unité d'Enregistrement Physique sur disque le secteur (128 mots)
  - Informations Système ; Secteurs de la Table des Fichiers, Secteurs de Table d'index, etc...
  - Informations Utilisateur non alignées aux frontières de secteurs.
- 2°) - Les IOCB (échanges par IOCS)
- 3°) - Les WCB (Mémoires de traitement d'une requête)
- 4°) - Les DF (descripteurs des fichiers ouverts)
  - Les TLG en Mémoire Centrale.
  - Les FAU (unité d'accès à un fichier)
- 5°) - Les Buffers de Travail (BT) pour les FAU publiques dans les systèmes qui traduisent les tâches esclaves en mémoire centrale (DRPS).

Les zones avec attente si tous les pavés sont alloués.

- ZUEP
  - $x_1$  pavés de 128 mots
  - Numéro de zone = 0
- ZIOCB
  - $x_2$  pavés de 8 mots
  - Numéro de zone = 1
- ZWCB
  - $x_3$  pavés de 50 mots, zone n° 2

Les zones avec paramètre de retour indiquant si la ressource a pu être allouée ou non.

- ZBT8, ZBT9, ZBT10 - Trois zones dont la taille et le nombre de pavés sont définis en fonction des besoins de l'application.
- ZDF
  - $x_4$  pavés de 25 mots.
  - Numéro de zone = 3
- ZUEP6, ZUEP7  $x$  pavés de  $2^i$  mots  $i$  de 7 à 11.

Remarque : Les valeurs  $x_1$   $x_2$   $x_3$   $x_4$  sont définies par l'utilisateur à la génération du Système d'Exploitation ainsi que L8, L9, L10 et  $X_8$ ,  $X_9$ ,  $X_{10}$ .

Les Sous-programmes de l'allocateur de Pavé :

### 1°) Allocation - Désallocation de pavés

- \* GETP - Permet de demander 1 pavé dans une zone. Avec RTES/D les pavés de taille inférieure ou égale à 20 mots sont remis à zéro.
- \* FREEP - Permet de libérer 1 pavé dans une zone.

### 2°) Chaînage de pavés

- \* PVIN - Insertion d'un pavé dans une liste, derrière un autre pavé.
- \* PVOUT - Déchaînage d'un pavé de sa liste.



c) Interface : FMS-E / GETP

Appel : CALL GETP ;

ENTREE (après le CALL)	SORTIE (avant RSR)																						
<p>RA : Numéro de zone (0, 1, 2 ou 3)</p>	<p>0 @ du pavé</p> <p>RA : selon carry</p> <p>1 RA : 0 (plus de pavé) RA : 1 (Numéro zone erroné)</p>																						
RB : "	RB : inchangé																						
RX : "	RX : inchangé																						
RY : "	RY : inchangé																						
RC : "	RC : inchangé																						
RL : "	RL : inchangé																						
RW : "	RW : inchangé																						
RS : "	<p>RS : carry 0 =&gt; pavé alloué</p> <p>1 =&gt; pavé non alloué</p>																						
<p>RK KSTORE de l'appelant</p> <p>→ <table border="1" style="width: 100%;"><tr><td style="text-align: center;">lien appelant</td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr></table></p>	lien appelant											<p>RK inchangé</p> <p>→ <table border="1" style="width: 100%;"><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr><tr><td> </td></tr></table></p>											
lien appelant																							

d) Interface : FMS-E / FREEP

Appel : CALL FREEP ;

ENTREE (après le CALL)	SORTIE (avant R S R.)
RA : @ du pavé à désallouer	RA : selon carry <div style="display: inline-block; vertical-align: middle;"> <div style="text-align: right; margin-right: 10px;">↗ 0 inchagé</div> <div style="text-align: right; margin-right: 10px;">↘ 1 détruit</div> </div>
RB : "	RB : inchagé
RX : No de la zone dans laquelle (est censé) se trouver le pavé	RX : inchagé
RY : "	RY : inchagé
RC : "	RC : inchagé
RL : "	RL : inchagé
RW : "	RW : inchagé
RS : "	RS : carry 0 → pavé désalloué 1 → pavé désalloué
RK : KSTORE de l'appelant	RK : inchagé
<div style="border: 1px solid black; padding: 2px; display: inline-block;">lien appelant</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block;">lien appelant</div>

e) interface : FMS-E / PVIN

Appel : CALL PVIN ; << (appel masqué)

ENTREE (après le CALL)	SORTIE (avant END.)
RA : @ du pavé à chaîner	RA : inchangé
RB : @ du pavé amont	RB : inchangé
RX : "	RX : inchangé
RY : "	RY : inchangé
RC : "	RC : inchangé
RL : "	RL : inchangé
RW : "	RW : inchangé
RS : "	RS : inchangé
RK : KSTORE de l'appelant	RK : inchangé

f) Interface : FMS-E / PVOUT

Appel' : CALL PVOUT ; << (appel masqué)

---

ENTREE (après le CALL)	SORTIE (avant END.)
RA : @ du pavé à déchaîner	RA : inchangé
RB : "	RB : inchangé
RX : "	RX : inchangé
RY : "	RY : inchangé
RC : "	RC : inchangé
RL : "	RL : inchangé
RW : "	RW : inchangé
RS : "	RS : inchangé
RK : KSTORE de l'appelant	RK : inchangé

#### 2.2.6. - Interface FMS-E / SSPFMS

Un quelconque module de FMS-E et en particulier une méthode d'accès peut utiliser les sous-programmes de service de FMS-E.

- VERFCB : Vérification FCB, table de codes d'arrêt.
- RADIX : transcodage ASCII - RADIX 40
- VERBUF : vérification zone mémoire (adresse, longueur).

- PARBUF : recherche d'un code d'arrêt.

Il est également fourni à titre indicatif l'interface d'appel du sous-programme BIDON du module OPEN-CLOSE.

Pour l'adressage 1024 K le module OPCGEN contient les sous-programmes de service suivant :

- LARBT : RA <- 1 mot du buffer de travail
- STAR BT : 1 mot du buffer de travail <- RA
- LAR BU : RA <- 1 mot de la zone d'échange
- MVMIL : Move 1024 K (double SLO).

**a) Interface : FMS-E / VERFCB**

**Fonction : Vérifier la validité de l'adresse fournie pour un FCB ou une table de code d'arrêt.**

**Appel : CALL VERFCB ;**

EN ENTREE	EN SORTIE
A : Adresse de zone (adresse du dernier mot)	A : détruit
B : "	B : détruit
X : 0 si esclave 1 si maître	X : inchangé
Y : "	Y : - 0 si OK - ADRFCB si non
C : "	C : Inchangé
L : "	L : Inchangé
W : "	W : Inchangé
K : Kstore de l'utilisateur	K : Inchangé
S : "	S : Inchangé (c, v détruits)

b) Interface : FMS-E / RADIX

Fonction : Compacter la zone alphanumérique d'un nom d'article (ANAM) ou d'un nom de fichier (FNAM - PUBW) en code RADIX 40.

Appel : CALL RADIX ;

EN ENTREE	EN SORTIE
A : "	A : détruit
= 0 radixer ANAM	
B : = 1 radixer FNAM-PUBW	B : détruit
X : "	X : détruit
Y : "	Y : compte rendu
C : @ W C B	C : inchangé
L : "	L : inchangé
W : @ D F	W : inchangé
K : Kstore de l'utilisateur	K : inchangé
S : "	S : inchangé (c, v détruits)
WANAM 1 } WANAM 2 } WANAM 3 } chargés WANAM 4 }	WTANAM 1 } WTANAM 2 } WTANAM 3 } codage en Radix 40

c) Interface : FMS-E / VERBUF

**Fonction** : Tester la validité de ABU (réajustée par rapport à SLO - SLE si mode esclave) et ABU + LBU. SI WSLOBU < 64 K, WABU est converti en adresse absolue et WSLOBU est mis à zéro.

**Appel** : CALL VERBUF ;

EN ENTREE	EN SORTIE
A : longueur du buffer en mots	A : inchangé
B : "	B : inchangé
X : "	X : inchangé
Y : "	Y : compte rendu 0 si OK, SYNTAX si non
C : adresse du WCB	C : inchangé
L : "	L : inchangé
W : "	W : inchangé
K : Kstore de l'utilisateur	K : inchangé
S : "	S : inchangé (c, v détruits)
WSLOBU } WSLEBU } inutilisés	WABU } WSLOBU } éventuellement rajustés
WABU : adresse du buffer	
MESCI : LBU en mots	



**d) Interface : FMS-E / PARBUF**

**Fonction :** Rechercher des codes d'arrêt dans une zone d'octets.  
REQUEST PORTCA ;

**Appel :** CALL PARBUF ;  
RELEASE PORTCA ;

EN ENTREE	EN SORTIE
A : "	A : nombre de mots CA compris
B : "	B : " "
X : longueur de la zone en mots	X : détruit
Y : "	Y : compte rendu
C : @ WCB	C : inchangé
L :	L : inchangé
W : "	W : inchangé
K : Kstore de l'utilisateur	K : inchangé
S : "	S : inchangé ( c, v détruits)

{  
 0 CA dans octet droit  
 1 CA non trouvé  
 -1 CA dans octet gauche

**WAZE :** @ relative  
**WSLOZE :** SLO-SLE correspondant  
**WSLEZE :** à la zone d'échange  
**WATCD :** @ table des codes d'arrêt

e) Interface : FMS / BIDON

**Fonction** : Capturer des requêtes adressées à une méthode d'accès non gérée par FMS-E.

**Appel** : CALL TABMA (RX) ; << RX = N0 METHODE D'ACCES  
ou  
CALL TOPCMA (RX) ; << RX = N0 METHODE D'ACCES

EN ENTREE	EN SORTIE
A : "	A : inchangé
B : "	B : O (travail standard sur PP) pour codage primitive précédente
X : No de Méthode d'Accès	X : inchangé
Y : "	Y : compte rendu qui indique que la méthode d'accès n'est pas gérée par FMS-E '602B
C : @ WCB	C : inchangé
L : local de OPCL (OPCLOC)	L : inchangé
W : "	W : inchangé
K : Kstore de l'utilisateur	K : inchangé
S : "	S : inchangé (c, v détruits)

f) Interface : FMS-E / sous programmes d'adressage 1024 K

Fonction : Load et Store 1024 K entre RA et les différentes zones gérées par FMS :

Appel CALL LARBU  
CALL LARBT  
CALL STAR BT

Entrée	Sortie
<p><u>LAR</u> RC = @ WCB RX = @ relative Autres registres inchangés</p> <p>SLO           Initialisés</p> <p>SLE</p> <p><u>STAR</u> RC = @ WCB RX = @ relative RA = contenu à écrire</p> <p>SLO } SLE }       Initialisés</p>	<p><u>L A R</u> RA = contenu lu</p> <p>Autres registres inchangés</p> <p><u>STAR</u></p> <p>Registres inchangés sauf ra ; 0</p>

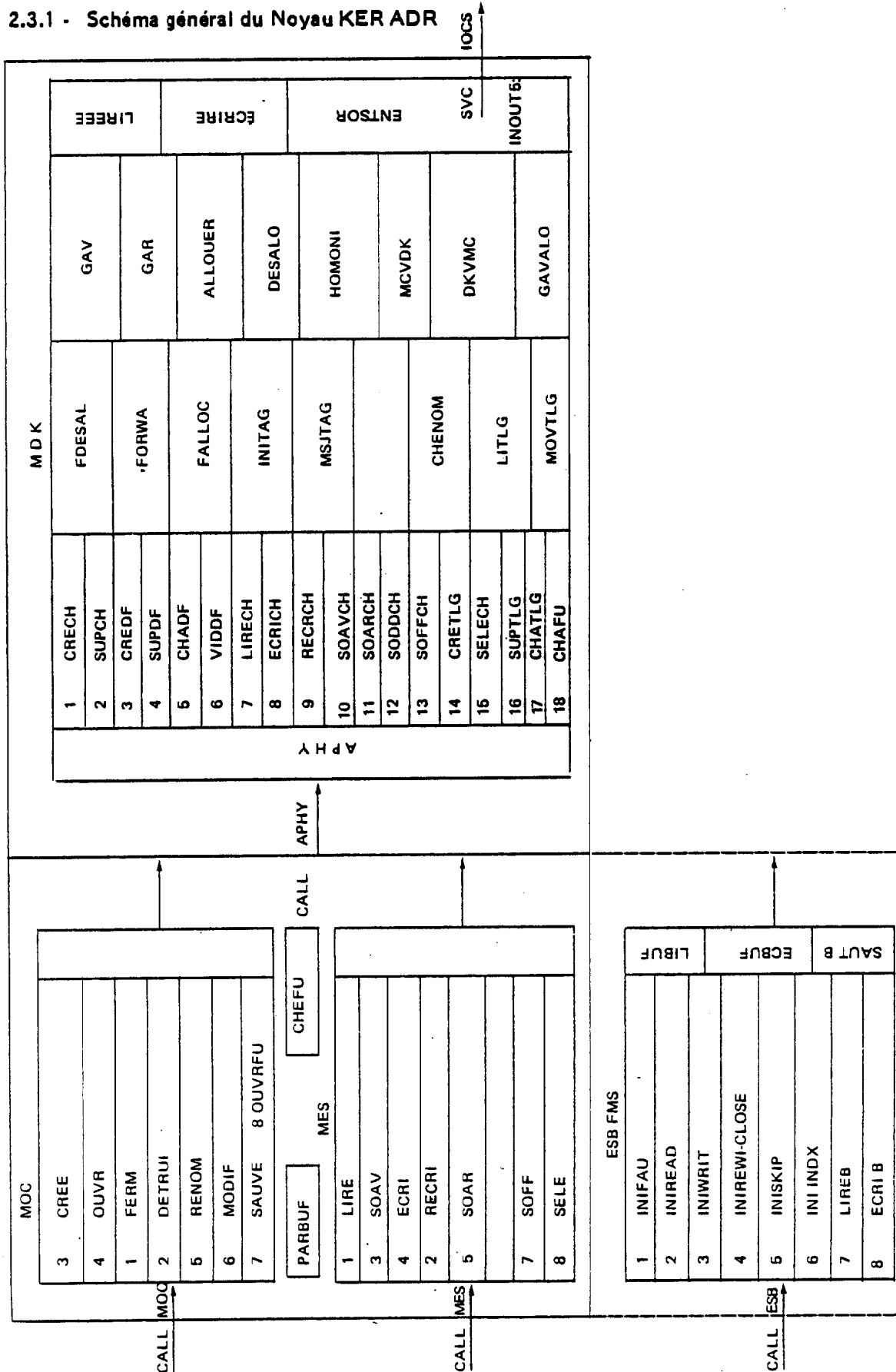
Fonction : Move 1024 K (Double SLO)

Appel : CALL MVMIL

Entrée	Sortie
<p>A : { -1 BT -&gt; BT       0 BT -&gt; ZE       1 ZE -&gt; BT</p> <p>B : " X : " Y : " C : @WCB L : " W : " K : Kstore</p> <p>WAORG : Adresse origine WADEST : " destination WLG MV : longueur du Move</p>	<p>A : détruit</p> <p>B : inchangés X : " Y : " C : @ WCB L : inchangé W : " K : "</p> <p>WAORG } WADEST }       détruits WLG MV }</p>

### 2.3. - LE NOYAU DE FMS-E

#### 2.3.1 - Schéma général du Noyau KER ADR



### 2.3.2. - Les primitives du Noyau : KERADR

Le noyau KERADR se distingue de KERFMS par les modules MOC et MDK.

- MDK : possède 3 primitives de plus CRETLG, CHATLG, SUPTLG
- MOC : est capable d'appeler ces 3 primitives.

Le module de bufferisation ESB FMS peut être associé à l'un des 2 Noyaux KERFMS ou KERADR.

Le Noyau est à lui tout seul un petit système de fichiers.

Les fichiers qu'il est capable de gérer, ne sont pas structurés en articles. Le Noyau gère donc ce que l'on appelle la structure de base d'un fichier dans le Manuel de Référence, chapitre 2.1.2.

#### a) Le module MDK

Le module MDK gère les chaînes de granules et les E/S physiques sur des chaînes de granules. C'est en particulier lui qui traite l'alignement secteur. Il fournit les primitives suivantes :

- |          |   |
|----------|---|
| * CRECH  | - créer une chaîne de granules                                  |
| * SUPCH  | - supprimer une chaîne de granules                              |
| * CREDF  | - créer un descripteur de fichier sur disque                    |
| * SUPDF  | - supprimer un descripteur de fichier sur disque                |
| * CHADF  | - chargement d'un descripteur de fichier en mémoire             |
| * VIDDF  | - mise à jour d'un DKDF dans le fichier des fichiers            |
| * LIRECH | - lire n mots sur une chaîne de granules                        |
| * ECRICH | - écriture de n mots sur une chaîne de granules                 |
| * RECRCH | - réécriture de n mots sur une chaîne de granules               |
| * SOAVCH | - saut en avant de n mots sur une chaîne de granules            |
| * SOARCH | - saut en arrière de n mots sur une chaîne de granules          |
| * SODDCH | - saut au début d'un fichier                                    |
| * SOFFCH | - saut à la fin d'un fichier                                    |
| * CRETLG | - créer la TLG dans le fichier (Organisation Physique Directe). |
| * SELECH | - transformer une adresse relative en absolue                   |
| * SUPTLG | - supprimer la TLG en mémoire                                   |
| * CHATLG | - charger la TLG en mémoire.                                    |
| * CHAFU  | - charger la TAG dans le DFU correspondant.                     |

Remarque :

- Dans le Noyau Standard KERFMS la primitive CRETLG se nomme en fait LITLG.
- APHY est un aiguillage dont le rôle est d'appeler les primitives du module MDK.

#### b) Le module MOC

Le module MOC gère les fichiers ouverts à un instant donné. Il fournit les primitives suivantes :

- \* CREE - créer un fichier temporaire (DF en MC)  
ou  
- créer un fichier permanent (DF sur Disque)
- \* OUVR - ouvrir un fichier permanent (lire DF)
- \* FERM - fermer un fichier permanent (réécrire DF)
- \* DETRUI - détruire un fichier permanent (détruire DF)
- \* RENOM - renommer un fichier permanent (modif DF)
- \* MODIF - modifier le type d'un fichier permanent (modif DF)
- \* SAUVE - transformer un fichier temporaire en permanent (copier DF sur Disque).
- \* OUVRFU - ouvrir une FU

#### c) Le module MES

Le module MES gère les E/S dans le récipient fichier (suite continue de n mots). Il réalise en particulier le traitement des codes d'arrêt, et fournit les primitives suivantes :

- \* LIRE - lire un paquet de mots en cpt de mots ou en code d'arrêt
- \* ECRI - écriture allocatrice ou désallocatrice en cpt de mots ou code d'arrêt
- \* RECRI - réécriture en cpt de mots ou code d'arrêt
- \* SOAV - sauter un paquets de mots en avant en cpt de mots ou code d'arrêt
- \* SOAR - sauter un paquet de mots en arrière
  
- \* SOFF - saut à la fin du fichier
- \* SELE - positionner PTC sur le nième mot du fichier

#### d) Le module ESB

Le module ESB gère les E/S dans le récipient fichier, en utilisant le buffer de travail fourni par l'utilisateur. Il traite également le code d'arrêt et fournit les primitives suivantes :

- INIFAU : initialisation de la bufferisation à la création d'une FAU
- INIREAD : initialisation su buffer avant lecture
- INIWRIT : initialisation du buffer avant écriture
- INIREWI-CLOSE : initialisation du buffer pour Rewind Close SKEOA EOJ

- **INISKIP** : initialisation de la bufferisation pour SKIPF SKIPB
- **INIINDX** : initialisation de la bufferisation pour IREAD IWRITE  
IRWRITE
- **LIREB** : lecture bufferisée
- **ECRIB** : écriture bufferisée

2.3.3. - Interface : OPEN-CLOSE / Noyau (MOC)

Appel : CALL MOC ; « RX := n° de primitive

ENTRÉE	SORTIE
RA : √	RA : détruit
RB : √	RB : détruit
RX : N° primitive	RX : détruit
	paramètre de retour
	* = 0 : OK
RY : √	* > 0 : erreur logique
	* 010 mot d'état PU erreur hardware
RC : @ WCB	RC : @ WCB
RL : local OPCLOC	RL : inchangé
RW : @ MCDF s'il existe	RW : détruit sauf CREE OUVR (@MCDF)
RS : √	RS : inchangé
RK : Kstore de l'appelant	RK : inchangé

Mémoires initialisés de WCB

- WTANAME1 nom du fichier
- WTANAME2 + catalogue
- WTANAME3 Radixé.
- WFUFTYP nom de la FU et PTYP du fichier
- WTART définition de la taille
- WNART initiale
- WNART1
- WFØNCT (ADR, NAR)



2.3.4. - Interface : Méthode d'accès / Noyau (MES)

Appel : CALL MES ; << RX : = n° de primitive

ENTREE	SORTIE
RA : "	RA : détruit
RB : "	RB : détruit
RX : n° de primitive	RX : détruit
RY : "	RY : { paramètre de retour * O £ Y £ N si = 0 : OK si > 0 : erreur logique * 010 mot d'état PU erreur hardware
RC : @ WCB	RC : @ WCB
RL : local de la méthode d'Accès	RL : inchangé
RW : @ MCDF	RW : détruit
RS : "	RS : inchangé
RK : Kstore de l'appelant	RK : Inchangé
<b>Mémoires positionnées de WCB</b>	<b>Mémoires initialisées en sortie</b>
- si accès séquentiel	
- WMESCI (= 40 si arrêt sur CA)	- WMESCØ compte de mots
- WATCD si table de CA/ 0 sinon	
- WAZU adresse zone échange	
- WSLOZE SLO zone échange	
- WSLEZE	
- si accès direct	
- WADR1 { adresse du nième	
- WADR2 { mot du fichier 31 bits	

**2.35 - Interface : Méthode d'Accès / ESBFMS**

**Fonction : Traitement de la bufferisation des E/S séquentielles.**

**Appel : CALL ESB ; << RX : = n° de primitive**

EN ENTREE	EN SORTIE
A : "	A : détruit
B : "	B : détruit
X : no de primitive	X : détruit
Y : "	Y : paramètre de retour
C : @ WCB	C : @
L : local de la méthode d'accès	L : inchangé
W : @ MCDF	W : détruit
K : Store de l'appelant	K : inchangé
S : "	S : détruit
<b>Mémoires utilisées</b>	<b>Mémoires initialisées en sortie</b>
- WMESCI : compte de mots	- WMESCØ compte de mots
- WATCD/0 : Adresse table code d'arrêt	
- WABU : Adresse de zone échange	
- WSLØBU, WSLEBU	

2.3.6. - Interface : MES, ESB / MDK

Appel : CALL APHY ; << RX : = n° de primitive

ENTREE	SORTIE
RA : "	RA : détruit
RB : "	RB : détruit
RX : N° primitive	RX : détruit
RY : "	RY : paramètre de retour
RC : @ WCB	RC : inchangé
RL : @ KERLOC	RL : inchangé
RW : @ MCDF	RW : @ MCDF
RS : "	RS : inchangé
RK : Kstore de l'appelant	RK : inchangé
 <b>Mémoire positionnées de WCB</b>	 <b>Mémoires initialisées en sortie</b>
- WMDKCI compte de mot	- WMDKCØ incrémenté
- WSLOZE WSLEZE	- WMDKCI décrémenté
- WAZE (syn WAZU)	- WMDKQ (ECRICH, CRECH)
- WSLOMDK mis à 0 en entrée du case OF	

2.3.7. - Interface : MOC / MDK

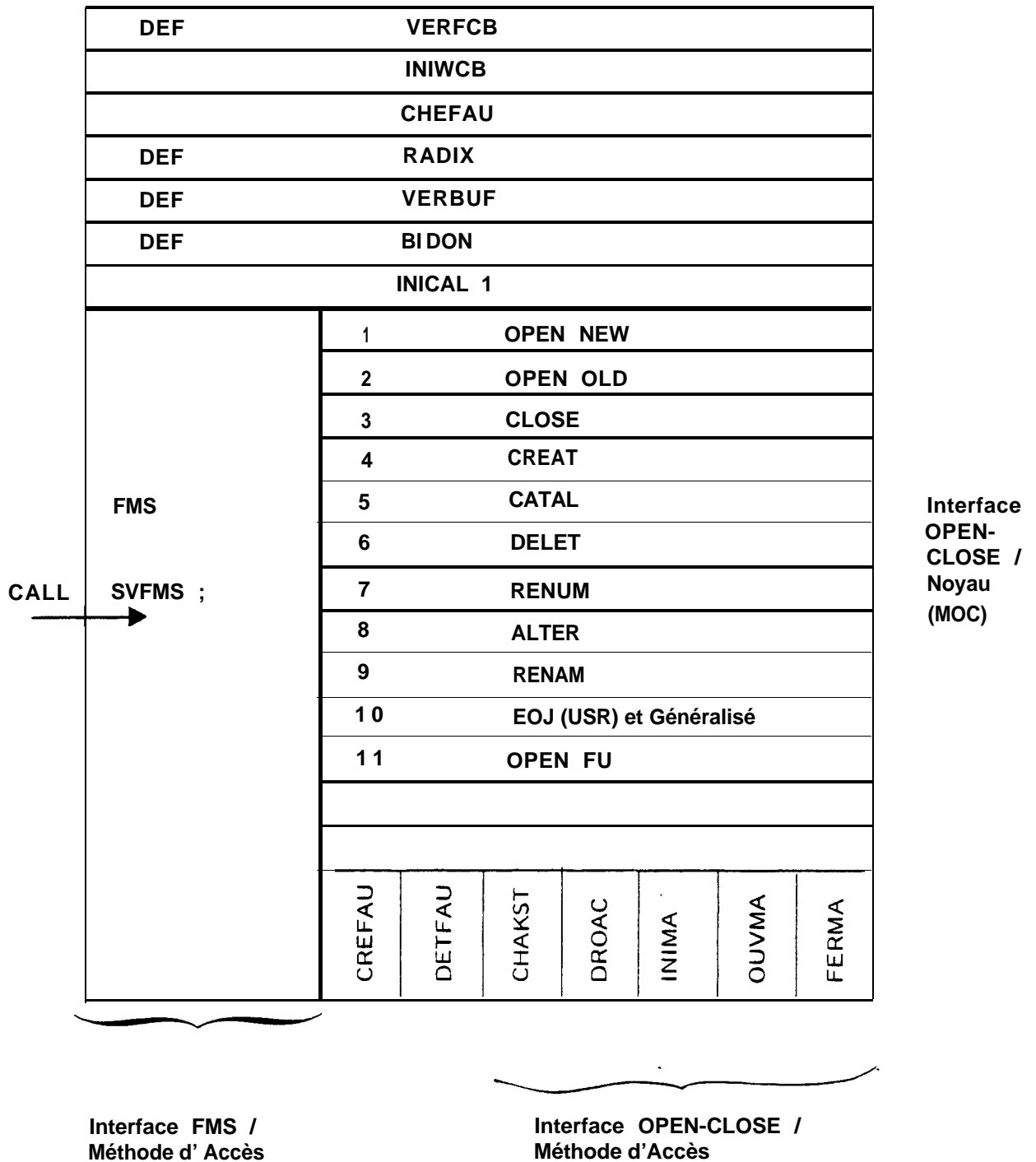
Appel : CALL APHY ( RX)

<< RX : = n° de primitive

ENTREE	SORTI E
RA : "	RA : détruit
RB : "	RB : détruit
RX : (No de la primitive de MDK)	RX : détruit
RY : "	RY : paramètre de retour
RC : @ WCB	RC : inchangé
RL : KERLOC	RL : inchangé
RW : @ MCDF	RW : @ MCDF
RS : "	RS : inchangé
RK : Kstore de l'appelant	RK : inchangé
Mémoires positionnées dans le WCB	Mémoires modifiées dans le WCB
- SUPCH, SUPDF, SODDCH : rien	WSAUV
- CREDF WTANAME 1	bit ident WEBOOLA
" 2	bit vide "
" 3	
WFUFTYP	
- CHADF WTANAME 1	
" 2	
" 3	
- VIDDF WTANAME 1	
" 2	
" 3	
bit HOMO de WEBOOLA	
- CRECH	
WTART	
WNART	
WNART1	

## 2.4. - LE MODULE OPEN-CLOSE

### 2.4.1. - Schéma général



## 2.4.2. - Les primitives du module OPEN-CLOSE

Le module OPEN-CLOSE c'est-à-dire OPCGEN, est constitué de 3 parties.

- FMS : le tronc commun d'entrée et de sortie de toutes les requêtes à FMS-E.
- l'OPEN-CLOSE proprement dit qui traite toutes les requêtes du niveau fichier.
- SSPFMS : un ensemble de sous-programmes de service utilisables par tout module de FMS-E.

C'est le module FMS qui appelle les méthodes d'accès pour le traitement des requêtes du niveau article ou portion d'article pour le Séquentiel. L'OPEN-CLOSE gère :

- Les Unités d'Accès (FAU) ouvertes à un instant donné sur des fichiers. C'est lui qui crée et détruit les FAU.
- Les droits d'accès aux fichiers (partage de ressources).
- L'appel des méthodes d'accès pour le traitement spécifique de création initialisation ouverture fermeture de fichiers.

### a) Les primitives OPEN-CLOSE

Il existe 4 classes de primitives.

- \* Classe des primitives de création de destruction d'ouverture de fermeture (5 primitives)
  - \* - CREAT 5 : création d'un fichier permanent (création d'une FAU)
  - DELET 6 : destruction d'un fichier permanent (détruire la FAU)
  - OPEN OLD 3 : ouverture d'un fichier permanent (création d'une FAU)
  - CLOSE 2 : fermeture d'un fichier permanent (détruire la FAU)
  - OPEN NEW 1 : création d'un fichier temporaire (création d'une FAU)
  - CLOSE 2 : destruction d'un fichier temporaire (détruire la FAU).

Remarque : il existe 3 primitives permettant la création d'une Unité d'Accès à un fichier et 3 seulement. Toutes les autres primitives du Système de Fichier s'y réfèrent.

- CREAT
- OPEN OLD
- OPEN NEW

- \* Classe des primitives de modifications statiques (4 primitives) et dynamiques de fichiers
  - RENUM 7 : - changer dynamiquement le FNUM (N° d'utilisation) d'un fichier permanent ouvert sur une FU ou d'un fichier temporaire créé sur une FU.
  - CATAL 5 : - transformer un fichier temporaire créé sur une FU en fichier permanent ouvert sur cette FU (transformation statique).



b) Primitives système

**NOM OPEN FU**

**BUT :** Cette requête a pour objectif d'agir sur les info systèmes associées à une FU-support, c'est à dire pour des processeurs en mode esclave.

- connaître l'affectation SU-FU
- tester si une FU-support est fermée
- charger la TAG en mémoire centrale (optionnel)
- fournir à l'utilisateur dans le FCB les informations de la DFU correspondante.

Appel : RA := @ FCB ; SVC (FMS) ou FMS = '38

FCB 0 1 2 3 4 7 8 13 14 15

0	TAG	0	0	0	' F F				
1					0				
2					0				
3					P R				
4	FU			←	ofu	BOOLU			
5					CPTR				
6					L TAG M				
7	N Z U E P			←	N S E C				←
8	TG sur 1 mot			←	/	→	SU / FU octet droit		
9					N B G L				
10					N B G				
11					0 / L D F				
12					0 / A D T F				

Description des paramètres :

a) En entrée

- TAG booleen tq si TAG = 1, FMS doit charger la TAG en MC dans la DFU correspondante sans modifier la valeur de CPTR.
- FONCT = 0 (n° requête OPEN FU).
- SU/FU Numéro spécifiant la FU à ouvrir.



b) en sortie

FMS charge après le PR la DFU correspondante voir Manuel d'utilisation pour la description des différents paramètres.

En particulier CPTR indique le nombre de fichiers ouverts sur la FU à cet instant là.

Compte rendu	valeur de PR	Signification
	0	Primitive correctement exécutée
	'6018	Incompatibilité primitive fichier : la FU n'est pas gérée par FMS mais l'affectation SU-FU est valide ; la requête est inefficace mais le FCB est chargé normalement.
	'6020	Zone de Pavés saturée.
	'6028	Erreur de syntaxe
	'6029	Adresse de FCB invalide
	'602A	SU FU invalide
Erreurs graves	'6034	Informations système invalides
	'6035	FU verrouillée par IOCS
	'4. . .	Erreur hardware

Nom :	EOJ Généralisé
But :	Détruire tous les temporaires et les FAU correspondantes connues de FMS. Appel en mode maître. Fermer tous les permanents et détruire les FAU correspondantes. Appel en mode maître.

Appel : RA = @FCB ; SVC (FMS) FMS = '38

FCB	0	3	4	7	8	15
	0	' B		"		
	"					
	"					
	P R					←

Compte rendu	Valeur de PR	Signification
	0	Primitive correctement exécutée
	'6028	Erreur de syntaxe
	'6029	Adresse de FCB invalide
	'602B	Méthode d'accès non gérée
Erreurs graves	'6032	Informations système invalides
	'6033	"
	'6034	"
	'6035	FU verrouillée par IOCS
	'4. . .	Erreur hardware.

Solar 16

**Nom :** CLEAR  
**But :** Réinitialisation des sémaphores de FMS par un programme en mode maître.

**Appel :** RA = @FCB ; SVC FMS FMS = '28

FCB	0	3 4	7 8	15
0	0	' C	"	
1			"	
2			"	
3			PR	←

Compte Rendu	Valeur de PR	Signification
	0	Primitive correctement exécutée

**Attention :**  
Le CLEAR ne réinitialise pas le booleen Pricou : Primitive en cours (bit 0 de BOOLA) de chaque FAU. Si nécessaire le faire à l'aide de ADRFAU.  
Un EOJ GEN doit normalement suivre la requête CLEAR.

**Nom :** ADRFAU  
**But :** Fournir à un programme en mode maître l'adresse d'une FAU et en option libérer cette FAU. (Reset du booleen primitive en cours : PRICOU (ou PC)).

**Appel :** RA = @FCB ; SVC (FMS) ; <<FMS = '38

FCB	0	1	3 4	7 8	15
	PC	0	' D	F N U M	
				"	
				"	
				PR	←
				@FAU	←

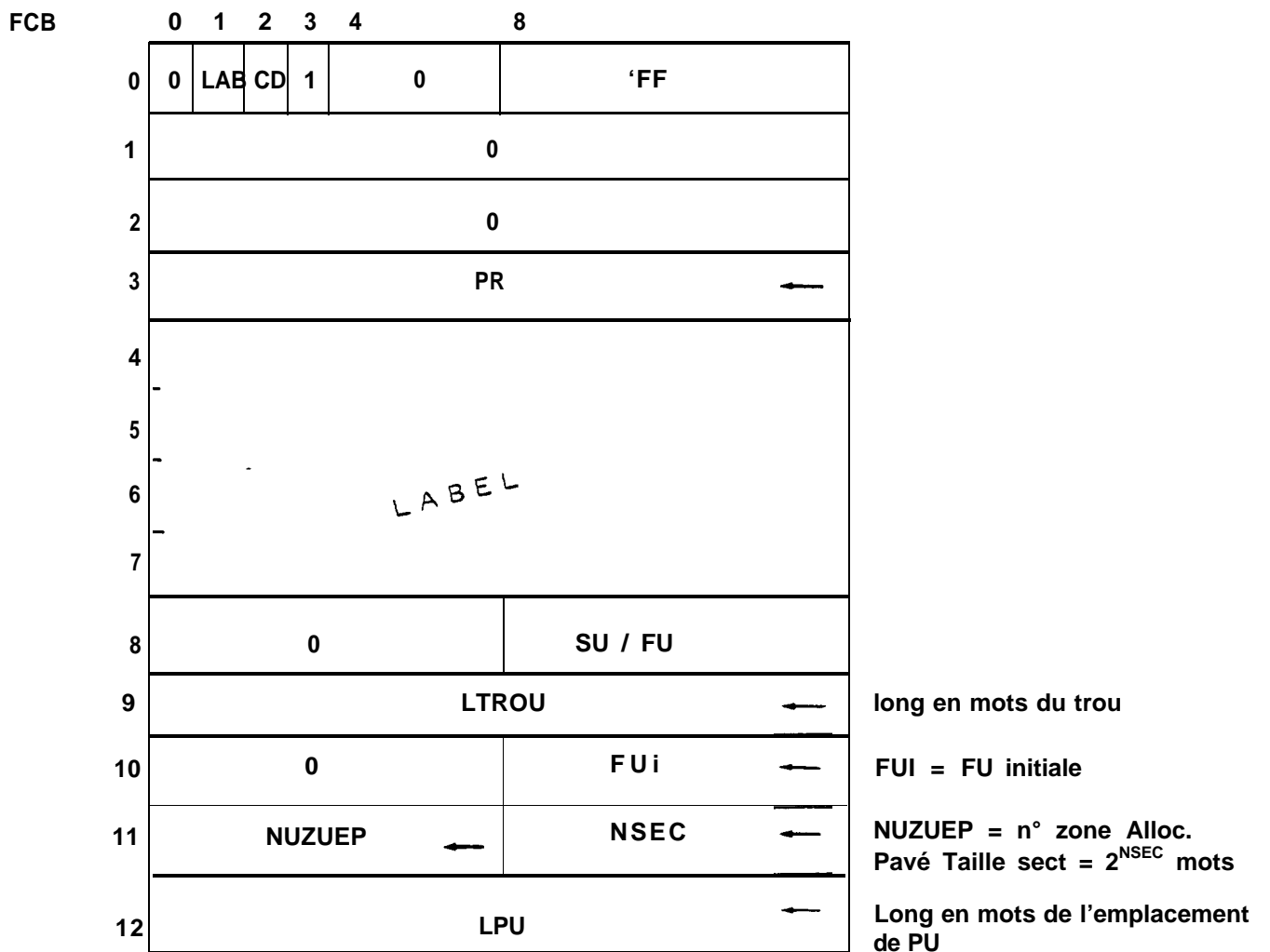
Compte Rendu	Valeur de PR	Signification
	0	Primitive correctement exécutée
	'600A	FAU inexistante
	'601F	Primitive en cours : Le FCB est chargé de plus Si PC = 0 FMS signale primitive en cours mais laisse le booleen PRICOU inchangé. Si PC = 1 FMS signale primitive en cours mais libère la FAU : Reset PRICOU.
	'6028	Erreur de syntaxe : (@ FCB, FONCT, FNUM)
	'6029	Adresse de FCB invalide.

Gestion de volume (OPC GEN, OPC PUB) FMS-E, FMS-G

**Nom :** CREATE FU

**BUT :** Créer pour un emplacement de PU l'ensemble des descripteurs des FU gérées par FMS sur ce support. L'emplacement de PU est désigné par la FU initiale de l'emplacement.  
SVC en mode maître réservée au superviseur.

**Appel :** RA = @ FCB ; SVC (FMS) où FMS = '38



Description des paramètres		CREAT FU
CD	CD= 1 CD= 0	signifie CREATE FU signifie DELET FU
LAB	booléen indiquant si le label est fourni par l'utilisateur dans le FCB si oui (LAB = 1) un contrôle de label sera effectué avec celui du support. LAB = 0 les mots 4 à 7 peuvent être quelconques.	
LABEL	8 caractères ASCII pairs (8 bits). Ce nom est constitué de 1 à 7 caractères suivis du caractère RC suivis de Nuls.	
SU/FU	Numéro du SU ou FU initiale permettant d'identifier l'emplacement de PU.	

Compte rendu	Valeur du PR	Signification	CREAT FU
	0	Primitive correctement terminée	
	6003	Article plus long : il n'y a pas assez de place pour reconfigurer toutes les FU du support. Si aucune FU n'a pu être configurée, la requête est ineffective. La TDFU n'est pas modifiée l'emplacement correspondant est donc soit vide, soit il contient la précédente initialisation. Si la requête est partiellement exécutée, la ou les premières FU de la structure sont correctement utilisables par FMS.	
	600C	Fichier inexistant : Erreur de label si le contrôle a été demandé. La requête est ineffective.	
	600E	Article inexistant : Il n'existe aucune FU gérée par FMS sur ce support la requête est ineffective.	
	6018	Disque non structuré ou erreur de génération : définition de la FU initiale.	
	601A	Erreur d'enchaînement : il n'y a pas de gestion de volume sur ce type de disque. La requête est ineffective.	
	601E	Fichier occupé : Il existe au moins un fichier ouvert sur l'une des FU précédentes de l'emplacement de FU correspondant. La requête est ineffective.	
	6028	Erreur de Syntaxe : (FONCT, "FNUM" + 'FF)	
	6029	Adresse de FCB invalide.	
	602A	SU ou FU non gérée par IOCS : Affectation SU - FU invalide ou numéro de FU non généré dans les tables d'IOCS.	
	602B	Méthode d'accès non gérée : branche d'overlay (CREDFU) non chargée.	
	6034	Information invalide en mémoire : Informations invalides dans le descripteur de PU. Règle contrôlée par GENFMS LPU-LDPU <sup>3</sup> 8 Compte-rendu du Read Structure tel que RA = 0 et PR = '6000. La requête est ineffective.	
	6035	FU verrouillée par IOCS : Compte-rendu du Read Structure tel que RA = '6000 soit version d'IOCS sans gestion de volume, ou encore RA = 0 et PR = '6002, '6004, '6006 c'est-à-dire disque non structuré (pas de FU initiale) ou encore volume démonté du point de vue IOCS. Taille secteur disque différente de la génération GENFMS. Pour cet emplacement de PU l'ensemble des FU gérées par FMS n'est pas un sous-ensemble de celles gérées par IOCS.  La génération de la DPU n'est pas effectuée avec une FU initiale.	
	'4XXX	Erreur Hardware : '4000 + mot d'état PU sur la lecture des des secteurs 3 ou 4 de la FU initiale.	

## CREATE FU

### Description fonctionnelle

- Ouverture de toutes les FU gérées par FMS sur ce volume sur l'emplacement de PU correspondant à la FU désignée.
- A condition que, s'il est demandé le contrôle de label soit OK, et que les fichiers soient fermés sur les précédentes FU du même emplacement.
- La gestion dynamique d'un emplacement de la TDFU fonctionne dans les cas suivants :
  - 1°) L'emplacement est vide (trou)
    - après initialisation vide de GENFMS
    - après la requête DELET FU
  - 2°) L'emplacement est occupé par un jeu de FU identique ou différent du moment que le compteur de fichiers ouvert est à zéro pour chaque FU.
    - après initialisation statique par GENFMS
    - sans requête DELET FU.

Solar 16

**Nom :** DELET FU

**But :** Détruire pour un emplacement de PU l'ensemble des descripteurs des FU gérées par FMS sur le support avant qu'il soit démonté. L'emplacement de PU est désigné par la FU initiale de l'emplacement. SVC en mode maître réservé au superviseur.

Appel : RA : = @ FCB ; SVC (FMS) ; ou FMS = '38

FCB

	0	1	2	3	4	8	15	
0	RAZ	LAB	CD	1	0	'FF		
1	0							
2	0							
3	PR						←	
4	L A B E L							
5								
6								
7								
8	0				SU / FU			
9	LTROU						←	
10	0				FU <sub>i</sub>			
11	NUZUEP				NSEC			
12	LPU						←	

} idem CREAT FU

Description des paramètres DELET FU

Même signification que pour CREAT FU

- RAZ = 0 DELET FU inefficatif si les fichiers sont ouverts
- = 1 DELET FU s'exécute même si des fichiers sont ouverts et sortie de l'erreur '601E.

Compte rendu	Valeur de PR	Signification
	0	Primitive correctement exécutée
	600C	Fichier existant : idem CREATE FU
	600E	Article inexistant : " "
	601A	Erreur d'entraînement : idem CREATE FU
	601E	Fichier occupé : il existe au moins un fichier ouvert sur l'une des FU du support la requête est ineffective
	6028	Erreur de syntaxe : (FONCT, "FNUM" 1 'FF)
	6029	Adresse de FCB invalide
	602A	SU ou FU non gérée par IOCS : idem CREATE FU
	602B	Méthode d'accès non gérée : " "
	6034	Informations invalides en mémoire : " "
	6035	FU verrouillée par IOCS : " "
	'4XXX	Erreur hardware : " "



## Solar 16

---

### DELET FU

#### Description fonctionnelle

- Destruction de toutes les DFU de l'emplacement désigné par SU/FU
- A condition que : s'il est demandé le contrôle de label soit CORRECT et que les fichiers soient fermés sur toutes les FU gérées par FMS sur ce support
- Si l'emplacement est déjà fermé pour FMS  
FMS rend PR = 0.

Gestion de volume

FMS-E, FMS-G

Nom	: SPCR FU
But	: Sous-programme de gestion de volume intégré dans le superviseur d'un Système d'Exploitation et réalisant les 2 requêtes FMS CREAT FU et DELET FU

Ce sous-programme utilise la TDFU déclarée en externe dans le module  
TABFMS - : S

Appel : CALL SPCR FU en mode maître réservé au superviseur.

Interface

en entrée

en sortie

RA = "  
 RB = 

FU	0
----	---

  
 RX = 

FONCT	FNUM
-------	------

  
 RY = @ WCB  
 RC = "  
 RL = GPCLOC  
 RW = @ FCB en mode maître

RA = PR interne  
 RB = "  
 RX = "  
 RY =  
 RC = @ WCB  
 RL = OPCLOC  
 RW = @ FCB

Réf. externes

TDFU Table des FU et PU  
 LGTDFU long: de la TDFU

PR interne  
 0 Primitive correctement exécutée  
 '4XXX Erreur hardware  
 '6000 Branche absente requête interdite  
 3, C, E, IE, 2A, 34, 35

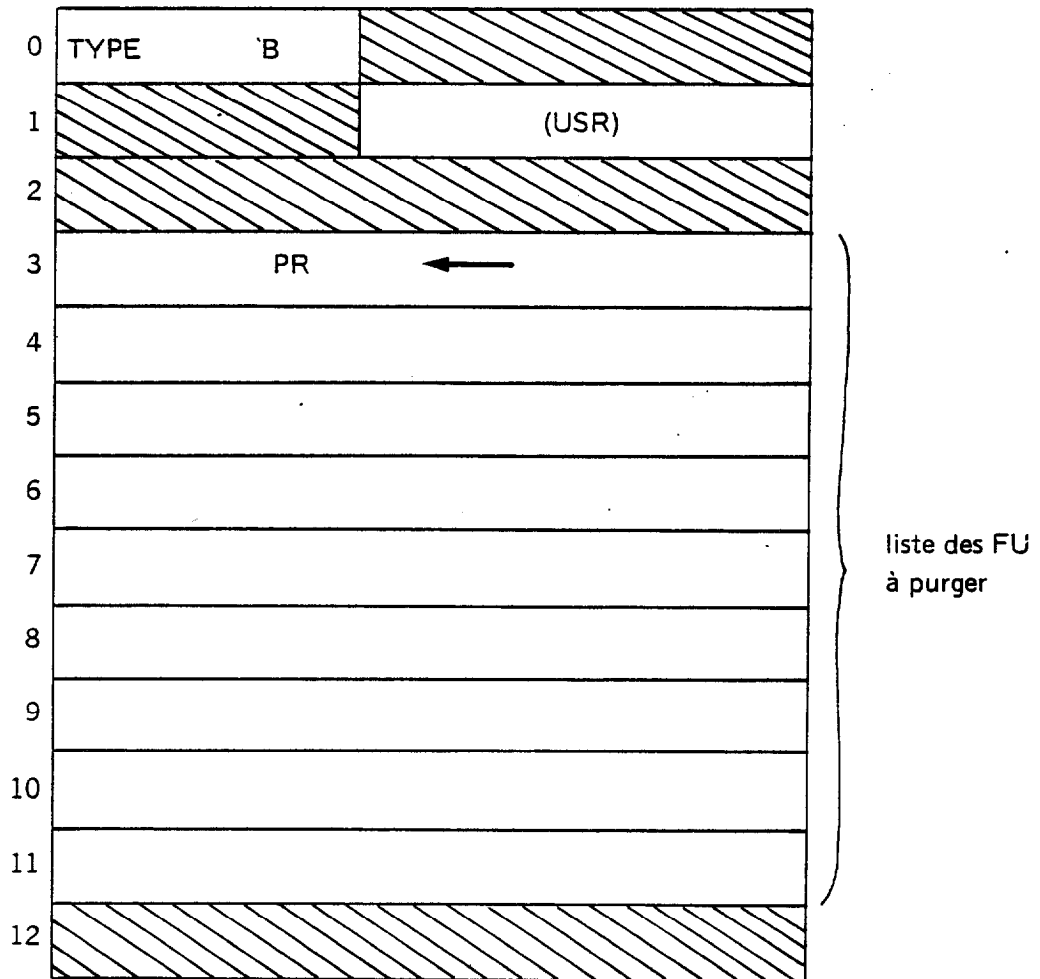
Sécurisation des données (OPCPUB)

**Nom** : PURGE

**But** : Obtenir la cohérence disque seul d'une base de données en écrivant sur le disque, les informations de la bufférisation restées en mémoire centrale.  
Les fichiers concernés ne sont pas fermés.  
La PURGE est totalement transparente à l'utilisateur.

Appel : RA := @ FCB ; SVC (FMS) <<FMS = '38

FCB



- TYPE 0 : EOJGEN
- TYPE 1 : PURGE généralisée publique avec sélection de FU (intéresse tous les fichiers ouverts en FAU publique)
- TYPE 2 : PURGE par utilisateur privé avec sélection de FU (intéresse tous les fichiers privés de l'usager spécifié dans l'octet droit du 2e mot du FCB).

**FCB mots 4 à 11** : file de bits représentant des FU  
La FU n° i est sélectionnée si le bit de rang i dans la file est à 1.

Compte rendu	Valeur de PR	Signification
	0	Primitive correctement exécutée
	'6028	
	'6028	Erreur de syntaxe
	'6029	@ FCB invalide
	'602B	Méthode d'accès non gérée
	'6032	Informations système invalides
	'6033	"
	'6034	"
	'6035	FU verrouillée par IOCS
	'4 . . .	Erreur hardware

**c) Codage de PP (Primitive Précédente)**

Chaque requête s'adressant à une Unité d'Accès, FAU doit indiquer dans l'octet PP quelle requête s'est déroulée et dans quelle condition.

- Codage de PP (mot 10 de la FAU)

0	3	4	7	8	11	12	13	15
N° M.A.		FONCT		N° Org. Logique		S	W	RWK

- PP
- le numéro de méthode d'accès est donné par le tableau d'identification des méthodes d'accès chapitre 2.5.1.
  - FONCT est en général celui du FCB associé à la requête.

Exemple OPEN OLD : PP = F2

Exceptions :

- PP invalide : FONCT : = 0
- PP pour le séquentiel :
  - READ Compte ('8X) => FONCT : = 1
  - WRITE Compte ('AX) => FONCT : = 3
  - READ Code d'arrêt ('CX) => FONCT : = 5
  - WRITE Code d'arrêt ('EX) => FONCT : = 7
  - SKIPB ('7C) => FONCT : = 'C
  - SKIPF ('7D) => FONCT : = 'D
  - REWIND ('7E) => FONCT : = 'E
  - SKEOA ('7F) => FONCT : = 'F

- Traitement de PP

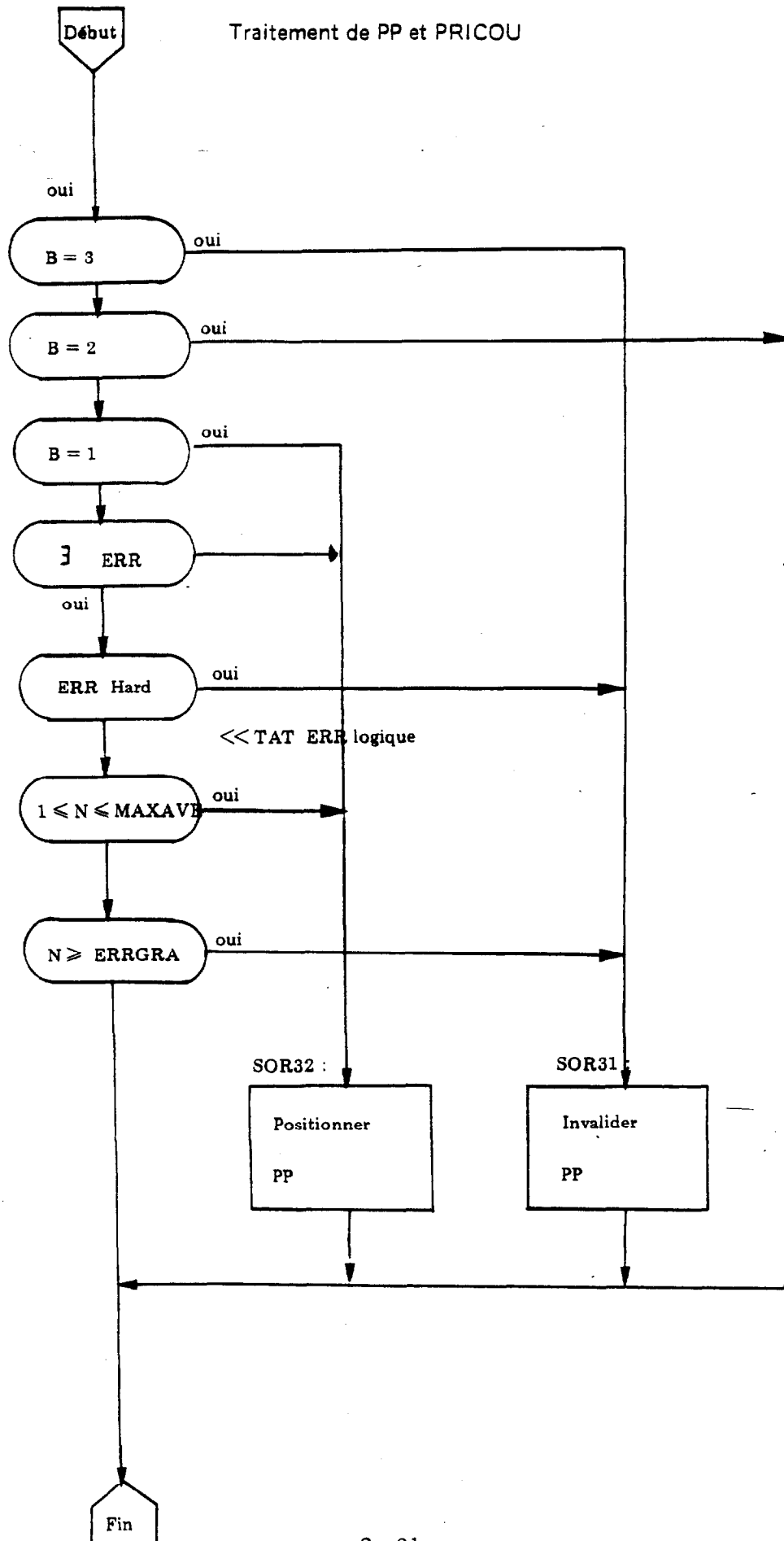
Selon le déroulement de la requête 3 traitements sont possibles :

- Positionner PP
- Laisser PP inchangé
- Invalider PP (FONCT : = 0)

- Conventions de codage de PP

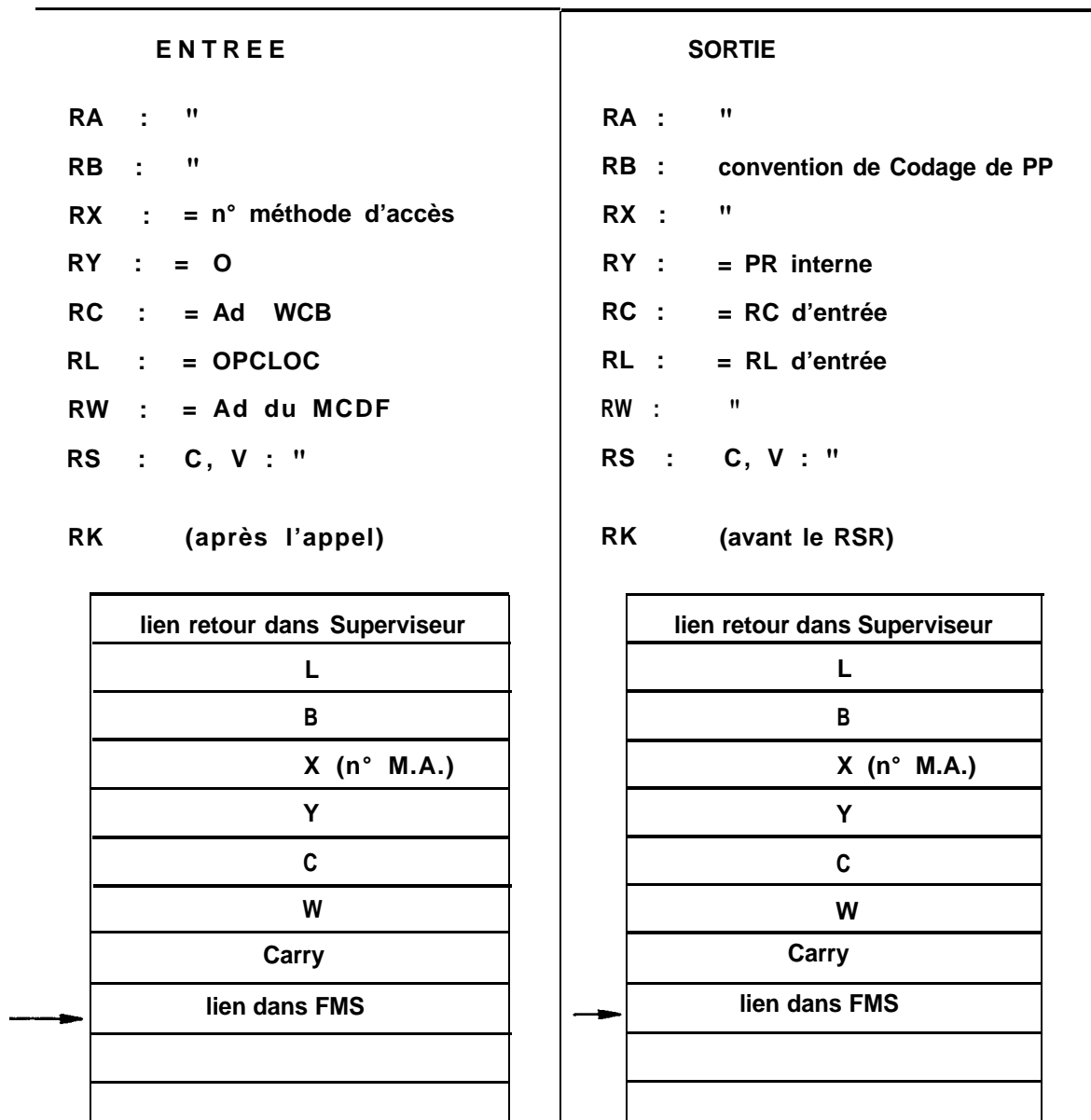
A la sortie de l'interface FMS / Méthode d'Accès il faut mettre dans le registre B une valeur indiquant à FMS quel traitement réaliser sur PP.

- B = 0 Traitement standard
  - OK ou 1 £ N £ MAXAVE => Positionner PP
  - MAXAVE < N < ERRGRA => inchangé
  - Erreur hardware ou N ≠ ERRGRA } => Invalider PP
- B = 1 Positionner PP
- B = 2 Laisser PP inchangé
- B = 3 Invalider PP



2.4.3. - Interface : FMS / Méthode d'Accès

Appel : CALL TABMA (RX) ; << RX : = n° de méthode d'accès







2.4.5. - Interface : FMS / OPEN-CLOSE

Appel : CASE (RX) / NBOPC OF << RX : = FONCT du FCB

ENTREE			SORTIE	
RA :	=	R X	RA :	= 0, 1, 2, 3, 4 Définition du Traitement de sortie
RB :	=	USR	RB :	= "
RX :	=	n° primitive OPC	RX :	= "
RY :	=	0 si EOJ USR 1 si EOJ Généralisé	RY :	= PR interne
RC :	=	@WCB	RC :	= @WCB
RL :	=	OPCLOC	RL :	= OPCLOC
RW :	=	@DF si \$ FAU	RW :	= MCDF
RS :	=	"	RS :	= C , V : "
RK			RK	

lien SVCH	lien SVC H	lien SVC H
L	L	L
B	B	B
X n° de requête	X n° de requête	X n° de requête
Y	Y	Y
C	C	C
W	W	W
Carry	Carry	Carry
USR	USR	
	flag	

lien SVC H
L
B
X (n° de requête)
Y
C
W (WFCB)
Carry

## 2.5. - UNE METHODE D'ACCES

Une méthode d'accès utilise 5 interfaces dans FMS-E.

2 interfaces d'appel de la méthode d'accès.

- interface FMS/Méthode d'Accès
- interface OPEN-CLOSE/Méthode d'Accès

3 interfaces par lesquels une méthode d'accès demande la réalisation d'un traitement.

- interface Méthode d'Accès/Noyau (MES)
- interface FMS-E/SSPFMS
- interface FMS-E/SUPFMS

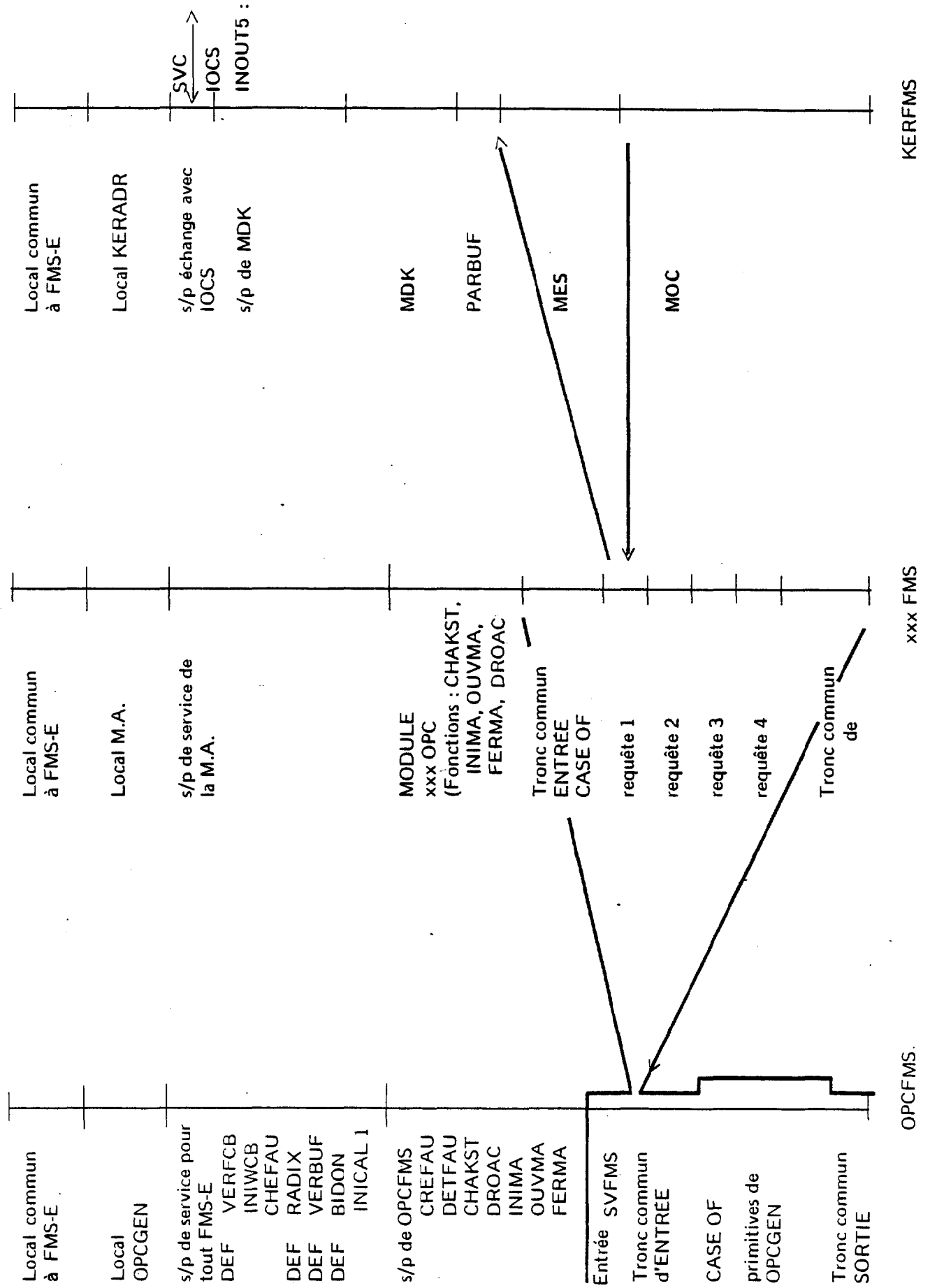
## 2.5.1. - Tableau général d'identification des Méthodes d'Accès

FONCTION	Nom de SVC	Numéro de Requête	Numéro de Méthode d'Accès	Nom de Méthode d'Accès	POROPC etc.
Traitement général FMS Traitement OPEN_CLOSE	FMS	' 3 8	- 1	OPC	POROPC etc.
Séquentiel	FMSS	* 3 9	0	SEQ	
Indexé	FMSI	* 3 A	1	IND	
Direct et Direct à Trous	FMSD	' 3 B	2	DIE DIIT	
Séquentiel Indexé	FMSX	' 2 8	3	SIX	
Séquentiel Chaîné	FMSC	' 2 9	4	SCH	
Direct Longueur Variable	FMSV	' 2 A	5	DIV	
		' 2 B	6		
		' 2 C	7		
		' 2 D	8		



- 2°) Une procédure xxx OPC qui effectue les fonctions d'Open-Close (CHAKST, INIMA, OUVMA, FERMA, DROAC) spécifiques à la M.A.
- 3°) Si le réalisateur le juge utile un ensemble de sous-programmes de service (procédures) utilisés par le module principal de la M.A. et par le sous-module xxx OPC.

**NB** : une M.A. doit être programmée en PL1600.



### 2.5.3. - Le Séquentiel

La méthode d'accès : Séquentiel permet 4 types d'accès séquentiels :

- Séquentiel PUR sur un fichier de type Séquentiel  
Dans ce cas le WRITE est Désallocateur-Allocateur (idem BM)  
Allocation Dynamique
- Séquentiel PUR sur un fichier de type Non Séquentiel  
Dans ce cas le WRITE est une réécriture dans le fichier  
Allocation Statique (non modification de la taille existante).

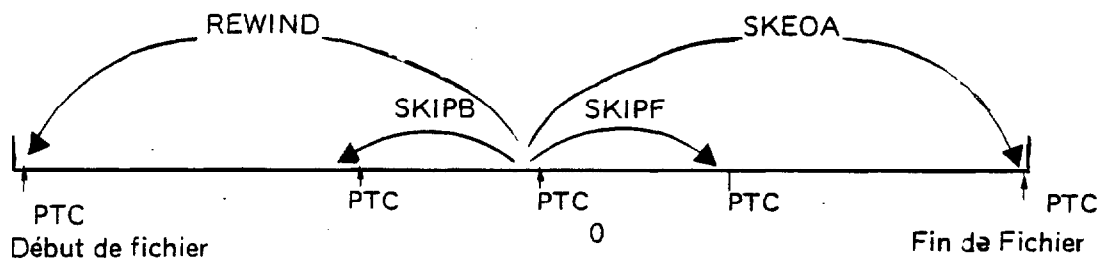
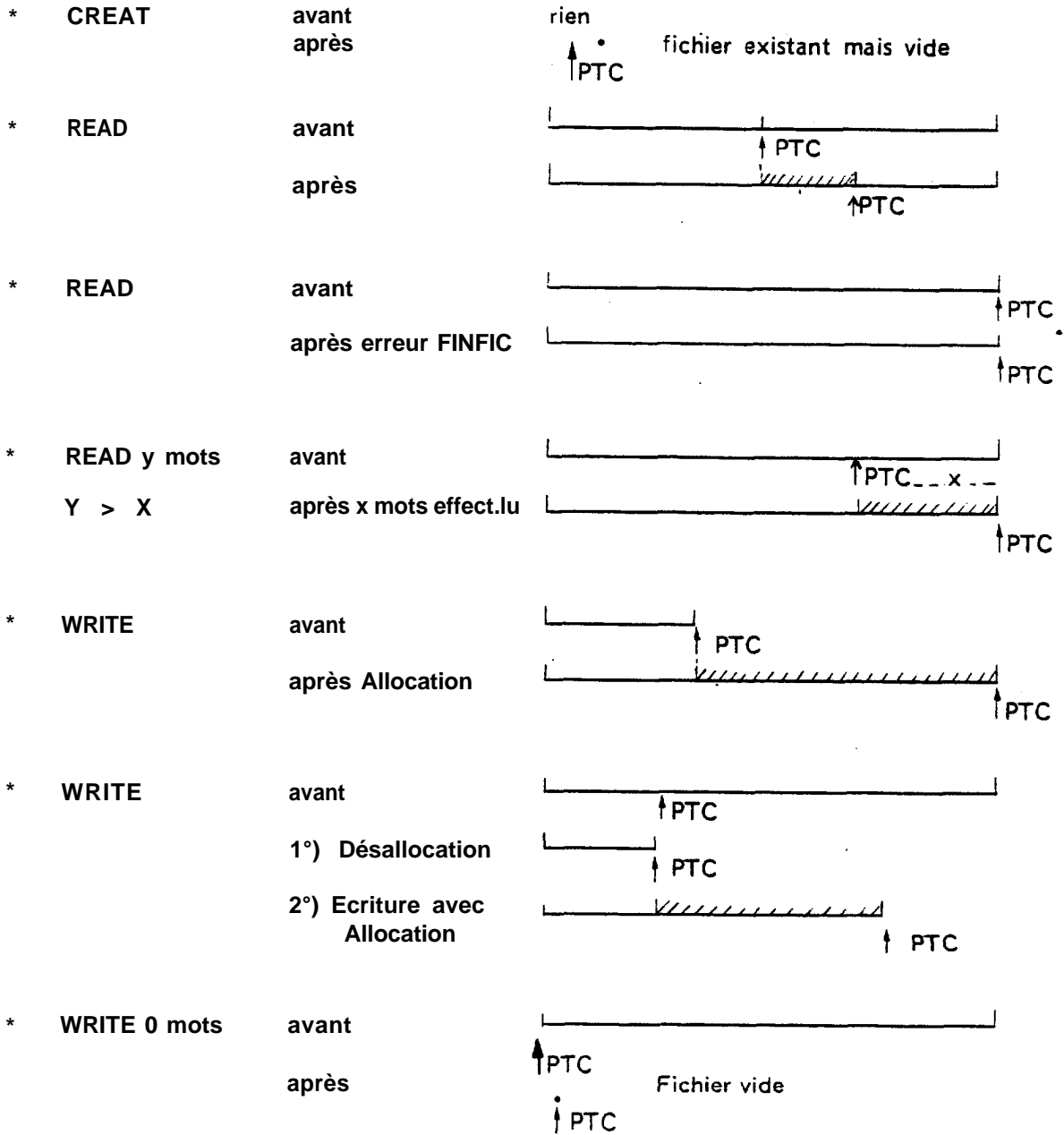
Cette possibilité n'est utilisable que :

- après un OPEN OLD
  - avant la première primitive valide de la Méthode d'Accès correspondant à celle du fichier.
- Exemple : avec un fichier Direct :

* OPEN OLD Read Write Skip Rewind DREAD (Invalide PR <sup>1</sup> 0) Read Write * DREAD (OK) Read Write	}	Séquentiel PUR
	}	Portion d'article

- Séquentiel sur Portion d'Article ; Alloué  
Une quelconque méthode d'accès peut définir un segment [A B] appartenant au fichier, à l'intérieur duquel les primitives de la méthode d'accès séquentielle sont utilisables.  
Dans ce cas WRITE est une réécriture dans l'article  
REWIND est un saut arrière au début de l'article  
SKEOA est un saut avant à la fin de l'article.  
  
=> Allocation Statique
- Séquentiel sur Portion d'Article ; Allocateur  
Une quelconque méthode d'accès peut définir un segment [A B] appartenant au fichier tel que la borne inférieure A est fixe, la borne B est modifiable à l'aide des primitives WRITE de la même façon qu'en séquentiel PUR allocateur, les autres primitives séquentielles permettant l'accès à l'intérieur de l'article :  
Dans ce cas WRITE est Désallocateur-Allocateur (idem BM)  
REWIND est un saut arrière au début de l'article  
SKEO est un saut avant à la fin de l'article.  
  
=> Allocation Dynamique

a) Séquentiel PUR (allocateur)



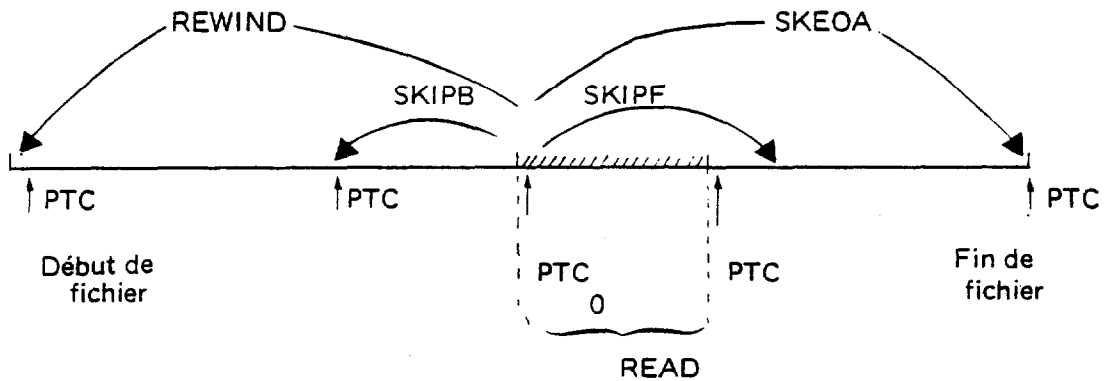


b) Séquentiel PUR Statique

Séquentiel PUR sur un fichier Non Séquentiel

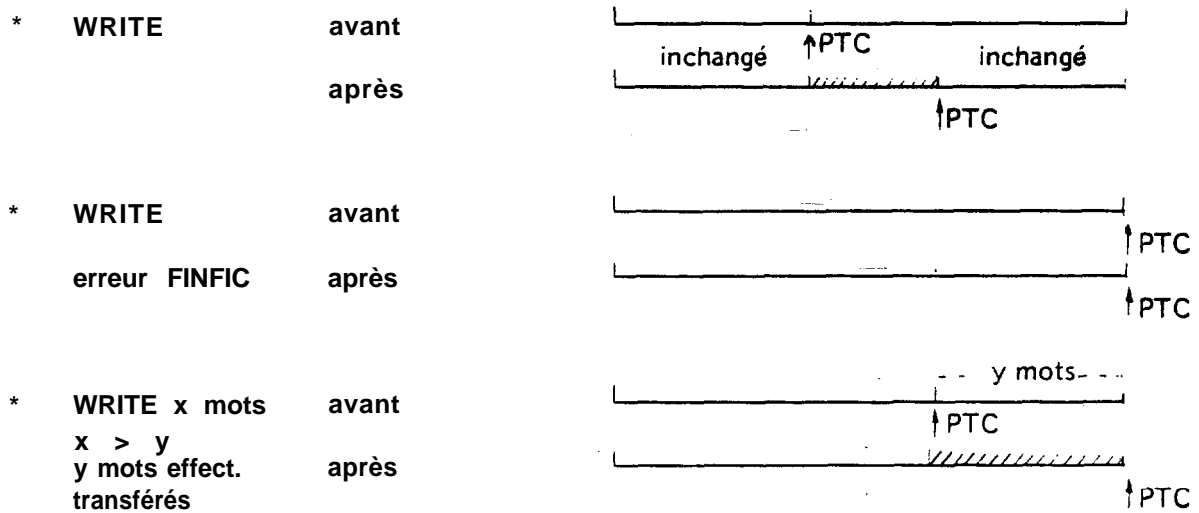
Après un OPEN OLD et avant toute primitive valide de la méthode d'accès correspondant à celle du fichier, le séquentiel PUR dit Statique est utilisable sur l'ensemble du fichier.

Les primitives : READ, REWIND, SKIPB, SKIPF, SKEOA sont identiques à celles du séquentiel PUR allocateur.



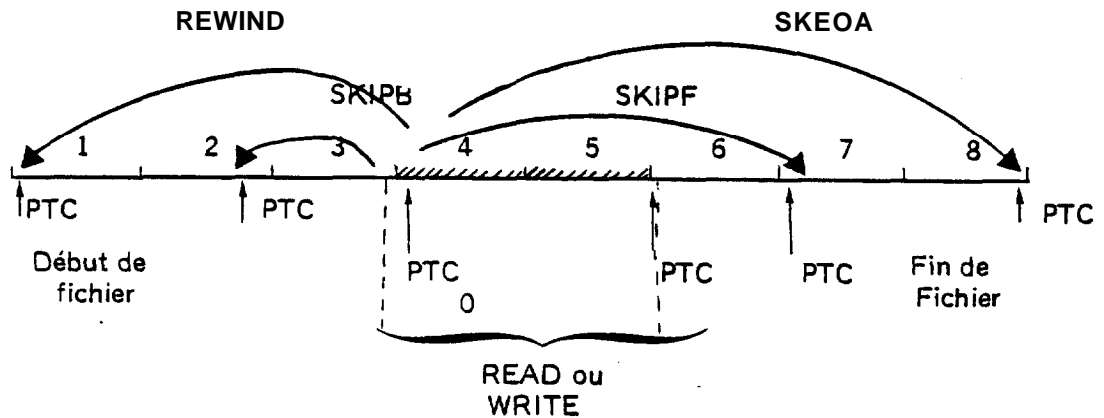
WRITE s'utilise comme un Read, de façon symétrique au read : Réécriture

WRITE = Réécriture



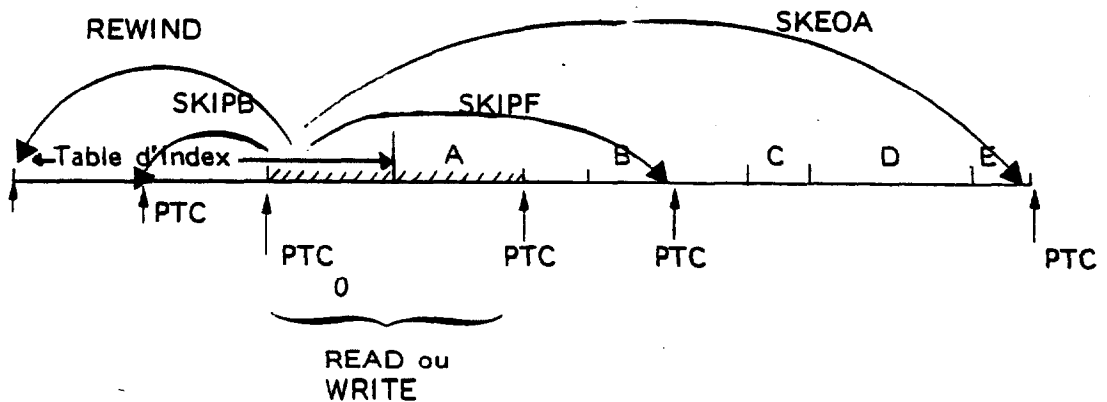
Séquentiel Statique : exemples

ex : Fichier Direct de 8 articles



La frontière des articles n'est pas reconnue en séquentiel PUR Statique.

ex : Fichier Indexé



De l'information système contenue dans le fichier peut être lue ou réécrite, détruite...

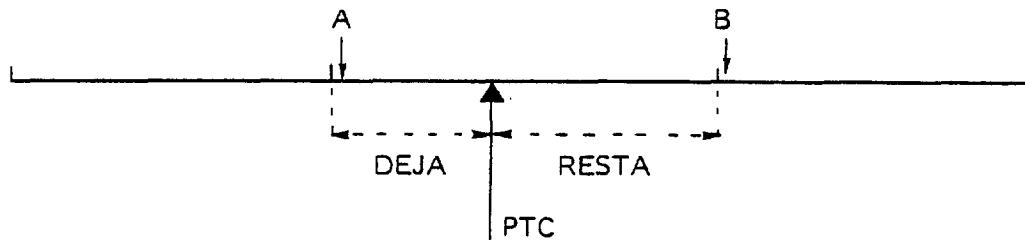
Après un OPEN OLD  
Après IRTIX

PTC est au début du fichier  
PTC est au début de l'information (1er article)

c) Portion d'Article Statique (PAS)

Séquentiel sur Portion d'Article existant

Le segment [A B] réservé à l'accès séquentiel P.A. est défini par rapport au PTC dans un fichier quelle que soit sa Méthode d'Accès.

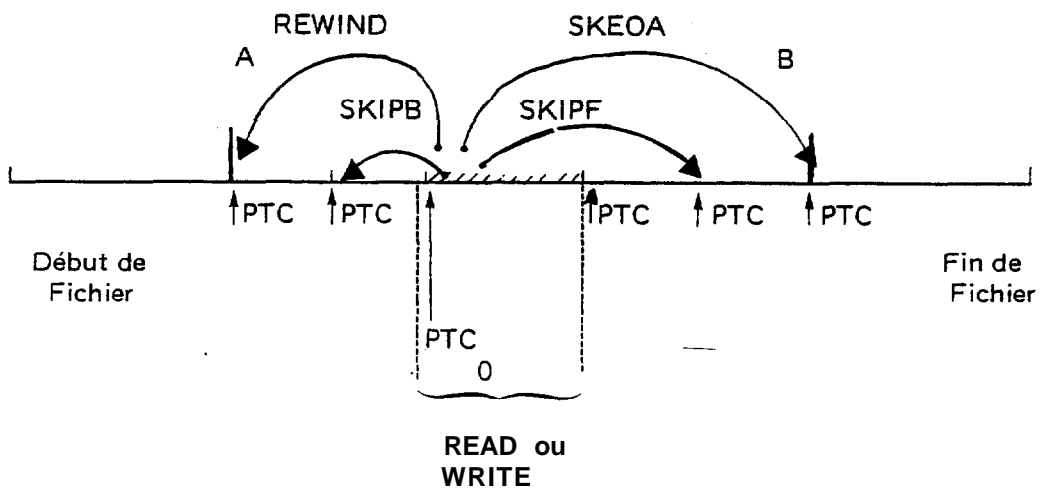


DEJA = nb de mots déjà traités dans le segment  
RESTA. = nb de mots restant à traiter dans le segment

RESTA + DEJA = Taille du segment

DEJA = 0 = borne inférieure de déplacement de PTC  
RESTA = 0 = borne supérieure de déplacement de PTC

Même gestion que pour le Séquentiel Statique, mais seulement sur le Segment [A B] au lieu d'être sur tout le fichier.

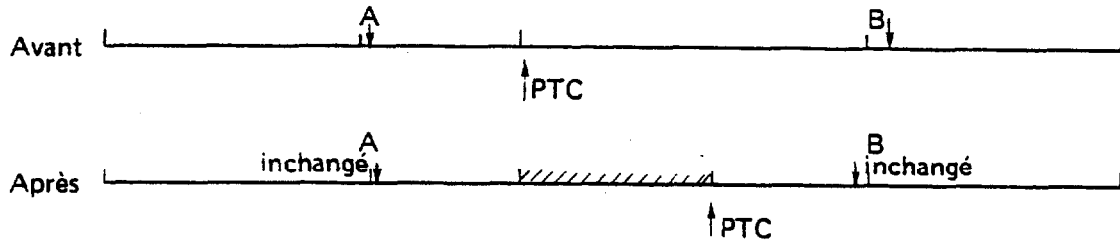


Dans chaque cas, MAJ de RESTA et DEJA par rapport au nouveau PTC.

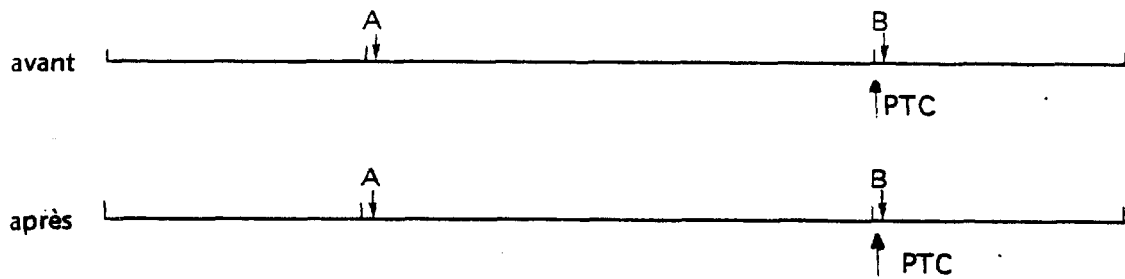
DEJA = DEJA + LBU  
RESTA = RESTA - LBU

échanger les signes si Backward.

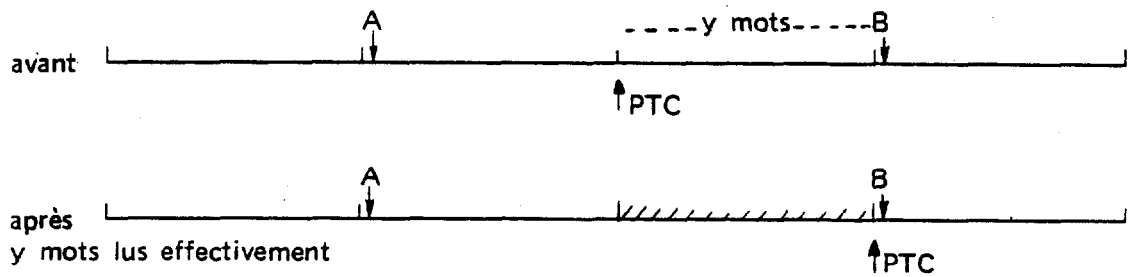
\* READ ou WRITE ou SKIPF (idem Read Write sauf transfert d'information)



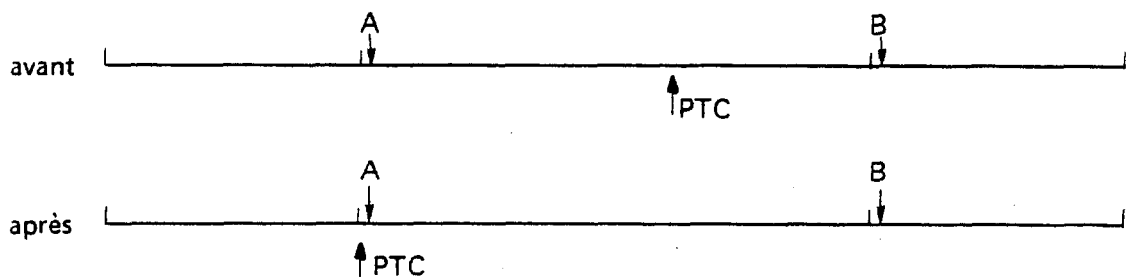
\* READ WRITE SKIPF (erreur FINFIC)



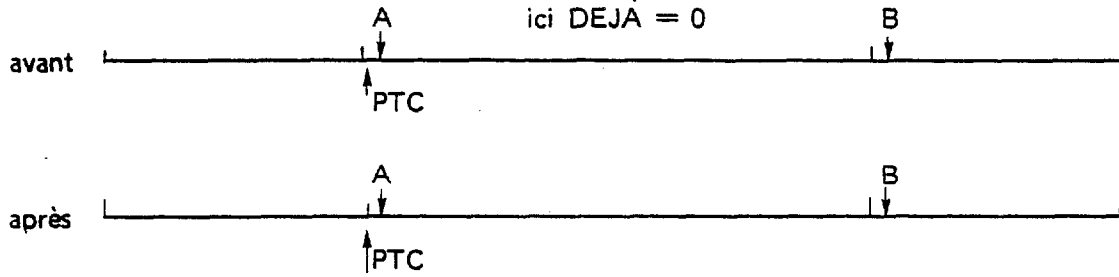
\* READ ou WRITE ou SKIPF de x mots tg  $x > y$



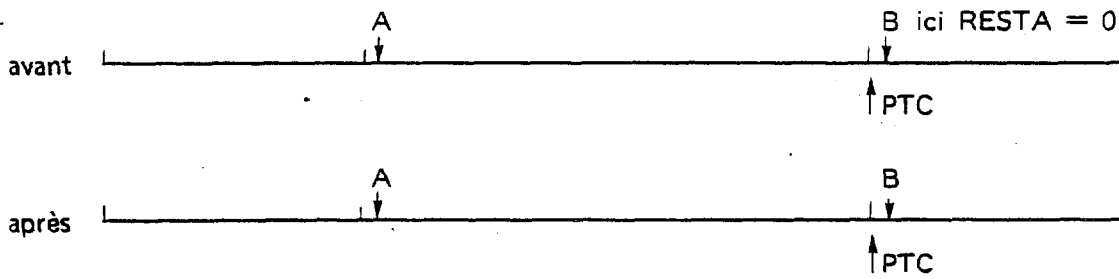
\* REWIND = saut arrière de Déjà mots



\* REWIND (PR = 0)



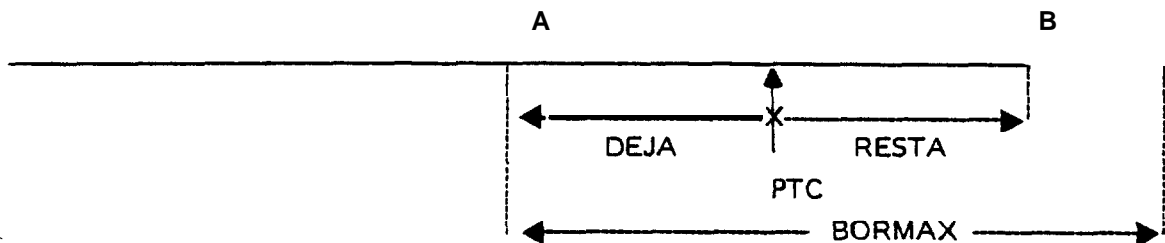
\* SKEOA = saut en avant de Resta mots (PR = 0)



## d) Portion d'Article Dynamique (PAD)

## Séquentiel sur Portion d'Article en cours de création

Le segment [A B] réserve à la P.A. allocatrice est défini par rapport au PTC dans un fichier quelle que soit sa méthode d'accès mais qui ne possède pas une organisation physique Directe. TLG dans le secteur zéro du premier granule du fichier.



DEJA = nb de mots déjà traités dans le segment

RESTA = nb de mots à traiter dans le segment

RESTA + DEJA = taille momentanée du segment

BORMAX = taille maximum du segment (FICTLO si erreur)

DEJA = 0 = borne inférieure de déplacement de PTC

DEJA = BORMAX, (RESTA = 0) = borne supérieure de déplacement de PTC

Même gestion que pour le séquentiel PUR pour le segment [A B]

READ, SKIPF, SKEOA (x mots) (SKEOA = saut avant Resta mots)

erreur FINFIC si RESTA = 0

DEJA = DEJA + x

RESTA = RESTA + x

SKIPB, REWIND (x mots) (REWIND = saut arrière Déjà mots)

erreur DEBFIC si DEJA = 0

DEJA = DEJA - x

RESTA = RESTA + x

WRITE (x mots) (ECRI)

1°) Désallocation

RESTA : = 0

2°) Allocation

DEJA : = DEJA + x

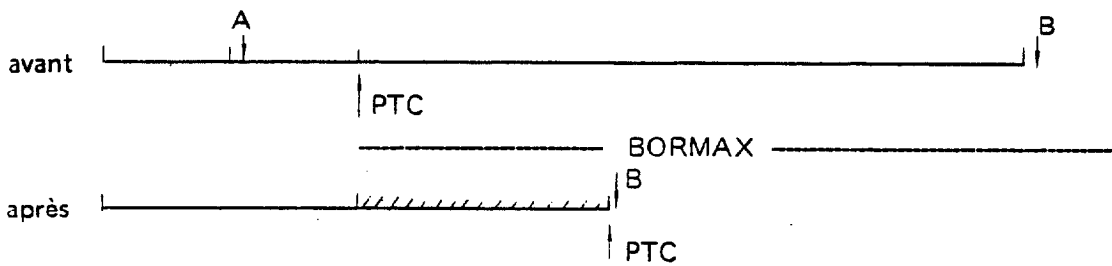
erreur FICTLO si DEJA > BORMAX

\* READ, REWIND, SKIPF, SKIPB, SKEOA

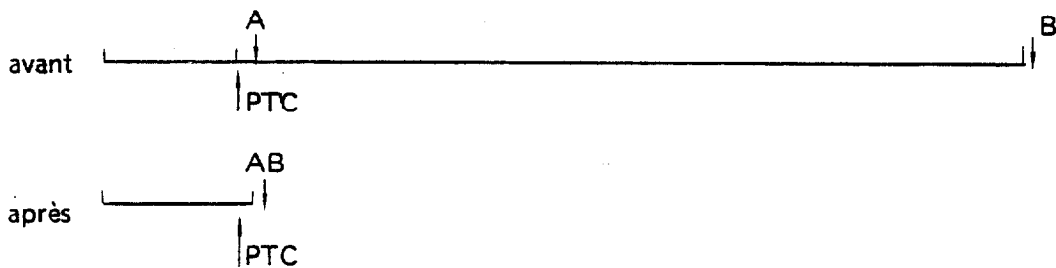
idem que P.A. Statique

- FINFIC
- DEBFIC
- compte effectivement traité

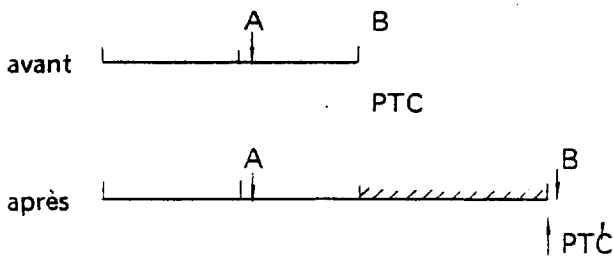
\* WRITE



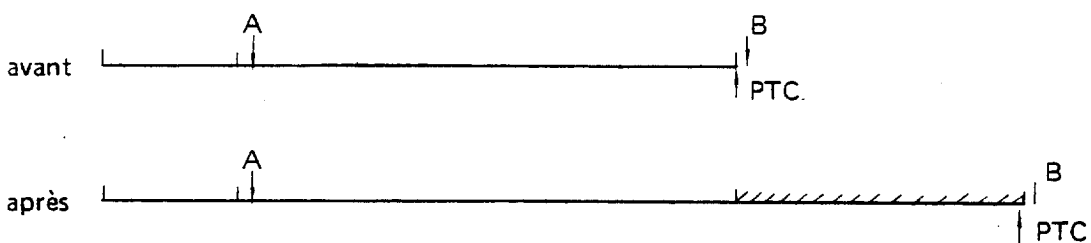
\* WRITE (0 mots)



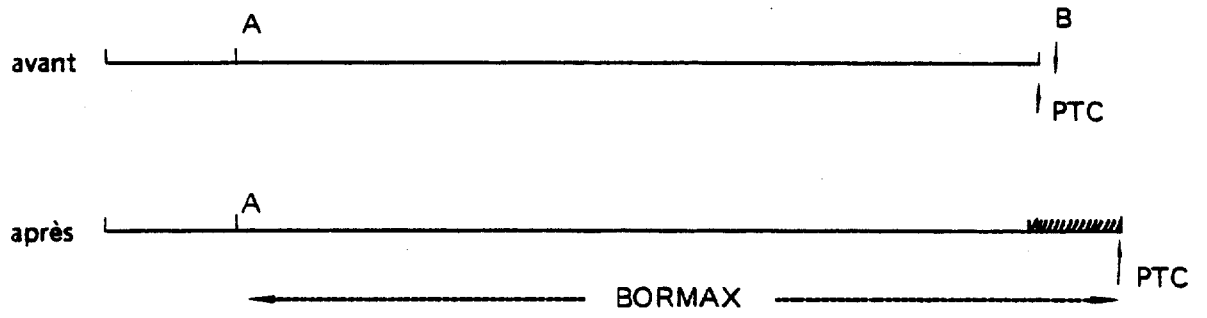
\* WRITE



\* WRITE



\* WRITE





### e) Commandes du Séquentiel

#### Booleens de commande du Séquentiel

<b>BORN</b>	:	<b>BORN</b>	=	<b>0</b>	<b>Séquentiel PUR</b>
		<b>BORN</b>	=	<b>1</b>	<b>Séquentiel sur Portion d'Article</b>
<b>S A</b>	:	<b>SA</b>	=	<b>0</b>	<b>Pas de sélection d'article</b>
		<b>SA</b>	=	<b>1</b>	<b>Sélection valide d'un segment pour la P.A.</b>
<b>WERE</b>	:	<b>WERE</b>	=	<b>0</b>	<b>Ecriture allocatrice désallocatrice WRITE = ECR</b>
		<b>WERE</b>	=	<b>1</b>	<b>Réécriture allocation statique WRITE = RECRI</b>

#### Booleens de Traitement

<b>UTILSA</b>	:	<b>UTILSA</b>	=	<b>0</b>	<b>La sélection d'article n'a pas été utilisée</b>
		<b>UTILSA</b>	=	<b>1</b>	<b>La sélection d'article a été utilisée</b>
<b>MAJTIX</b>	:	<b>MAJTIX</b>	=	<b>0</b>	<b>La taille de l'article (donc du fichier) n'a pas été modifiée</b>
		<b>MAJTIX</b>	=	<b>1</b>	<b>La taille de l'article (donc du fichier) a été modifiée (par un WRITE allocateur)</b>

#### Le module SEQ

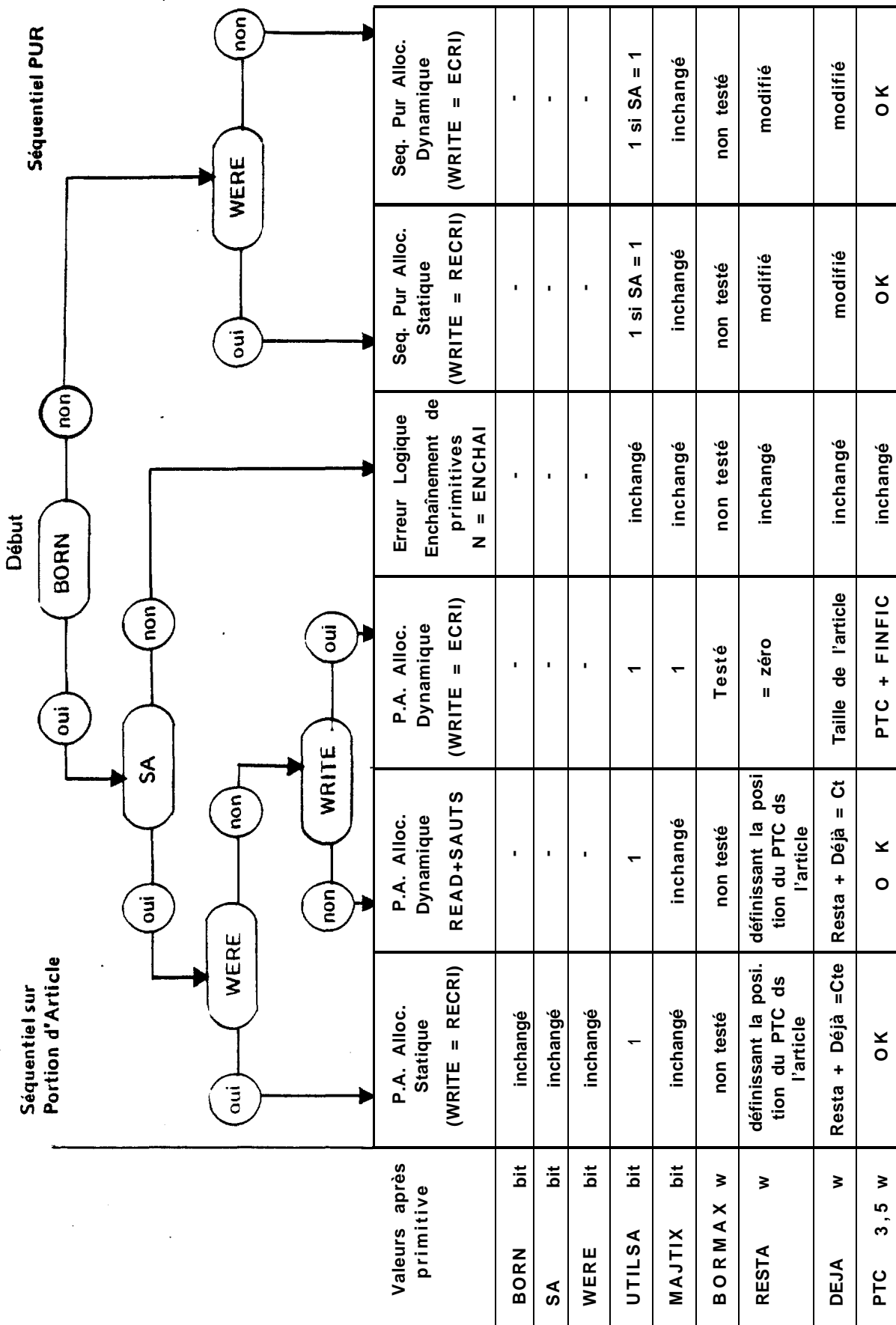
- Teste BORN, SA, WERE et les laisse inchangés
- MAJ de UTILSA et MAJTIX

#### Les autres méthodes d'accès

- Positionnent BORN, SA, WERE
- Peuvent tester UTILSA et MAJTIX et les réinitialiser

#### Le module OPEN CLOSE

- Initialise tous les booleens





### **3 - CONSEILS D'UTILISATION**

<b>3.1 - CONSEILS GENERAUX</b>	<b>3 - 2</b>
<b>3.1.1. - La taille des granules</b>	<b>3 - 2</b>
<b>3.2 - UTILISATION DES METHODES D'ACCES</b>	<b>3 - 5</b>
<b>3.2.1. - Le séquentiel indexé</b>	<b>3 - 5</b>



### 3 - CONSEILS D'UTILISATION

**Ce chapitre a pour but de fournir quelques conseils aux utilisateurs du système de fichiers FMS, et des utilitaires de gestion de fichiers FUP.**

### 3.1 - CONSEILS GENERAUX

#### 3.1.1 - La taille des granules

FMS fournit à l'utilisateur la possibilité de paramétrer sur chaque support disque et plus précisément sur chaque FU-support, la taille de l'unité d'allocation le granule.

Cette souplesse permet à l'utilisateur d'optimiser l'exploitation des fichiers disque en fonction de critères variés et parfois contradictoires.

La règle proposée est décomposée en un ensemble de règles élémentaires qu'il faut appliquer de façon globale et en fonction de contextes et d'objectifs différents.

##### 1°) Nombre minimal d'Unités Fonctionnelles

C'est dans ce cas en effet que l'allocation dynamique fournit le meilleur service vis-à-vis du taux d'occupation des disques.

Il y a lieu d'utiliser la notion d'Unité Fonctionnelle pour :

- mettre dans le même espace disque des fichiers liés entre eux logiquement.
- augmenter la performance des E/S sur fichier en optimisant sur chaque FU la taille des granules.
- augmenter la fiabilité en regroupant sur un même support les fichiers d'un usager.

##### 2°) Taille de granule maximale (TG)

Vis à vis de la place en mémoire centrale (ADR) et de la rapidité d'exécution des requêtes FMS (accès séquentiel) la performance maximale est obtenue avec des tailles de granules maximales. La seule contrainte étant la perte de place sur disque par rapport à la taille moyenne des fichiers, et à la taille des informations système, dans le cas de l'organisation standard.

$$TG = \sqrt{\frac{TFU}{i}} \quad \text{avec l'organisation standard}$$

Pour n'optimiser exclusivement que le taux d'occupation disque en fonction des informations systèmes gérées par FMS, il faut optimiser la place utile disponible soit P cette place utile.

TFU la taille de la FU en secteur.

TG la taille du granule.

NBG le nombre de granules (NBG = TFU/TG)

$$P = TFU - i.TG - (NBG-i)$$

i.TG représente l'occupation de FIFI constitué de i granules.

NBG-i représente la perte due au secteur système de chaque granule.

$$\text{Donc} \quad P = TFU - iTG - \frac{TFU}{TG} + i$$

Cette fonction P est optimale lorsque sa dérivée est nulle soit :

$$0 = P' = -i + \frac{TFU}{TG^2}$$

$$\text{d'où} \quad TG = + \sqrt{\frac{TFU}{i}} \quad (\text{Solution} > 0)$$

## Solar 16

Exemple : Pour une FU de 399 cylindres de 6 K mots et FIFI occupant 1 seul granule.

$$TG = \sqrt{\frac{TFU}{x}} = + \sqrt{\frac{19\ 152}{1}} \approx 138 \text{ secteurs}$$

- 3°) Petit fichier => petit granule  
Grand fichier => grand granule

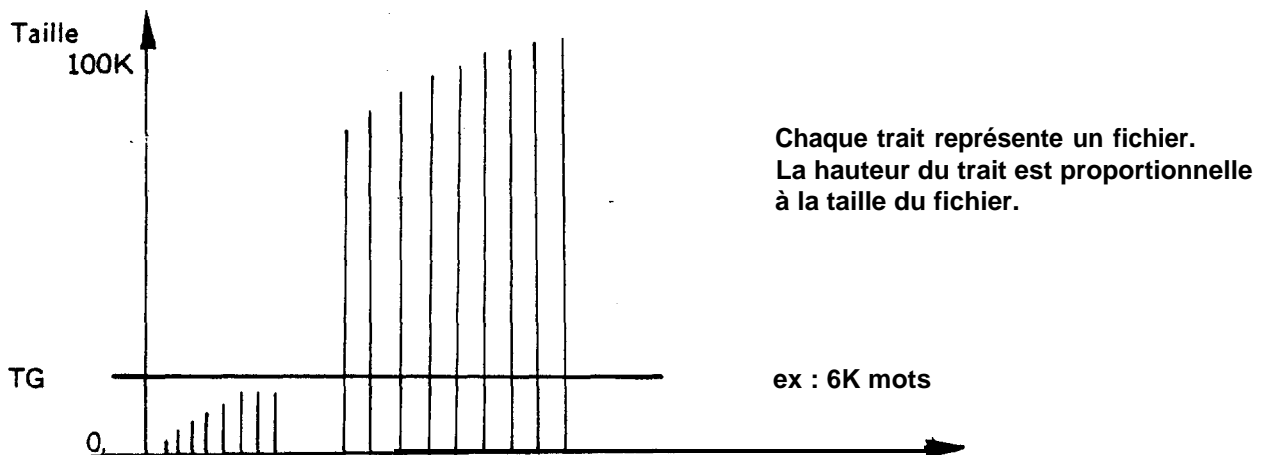
4°) 1 fichier occupe au moins 10 granules

D'un point de vue statistique la place perdue est de la moitié du dernier granule, soit une perte de 1/20, c'est à dire 5%.

Cette règle est facilement applicable lorsque tous les fichiers d'une FU-support sont de taille semblable.

5°) Cas particulier

La FU support contient des petits et des grands fichiers, sachant que en moyenne un grand fichier est 10 fois plus grand qu'un petit.

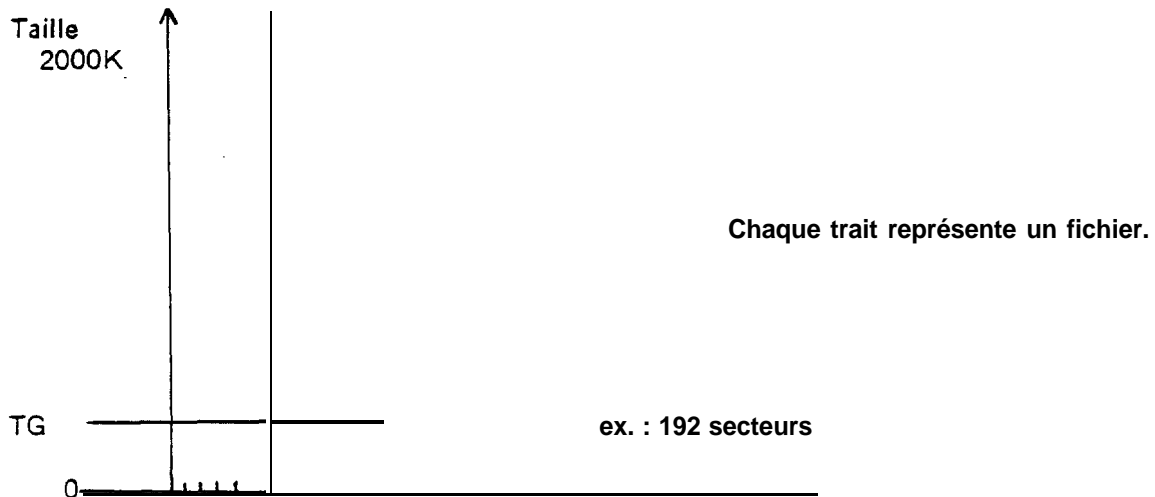


La plupart des petits fichiers tiennent dans un seul granule avec une perte de place négligeable. Les grands fichiers occupent un nombre de granules optimal (ni trop ni pas assez).



### 6°) Cas particulier

La FU support contient un fichier occupant 90 % de la FU et quelques petits fichiers.



La perte de place sur les petits fichiers est négligeable, il faut donc optimiser la taille des granules pour le ou les quelques grands fichiers par exemple TG de taille maximale ou  $\sqrt{\frac{TFU}{x}}$

### 7°) TG << limite IOCS

Sur chaque support IOCS limite la taille d'une entrée-sortie (ex. : 6 Kmots sur les disques à bras mobile 2,5 Mmots.)

Règles

Si TG < limite IOCS => une E/S FMS (LBU) peut être de taille quelconque (< 32 Kmots).

Si TG > limite IOCS => LBU < limite IOCS c'est à dire que l'utilisateur doit respecter la limite IOCS pour ses E/S FMS.

### 8°) TG moyen = 2 Kmots ou 16 secteurs

Lorsque l'on ne possède aucune information sur les conditions d'utilisation ou les critères d'optimisation, il faut choisir la taille moyenne 16 secteurs.

### 3.2 - UTILISATION DES METHODES D'ACCES

#### 3.2.1 - Le séquentiel indexé : calcul d'encombrement

Soit un fichier tel que :

N = nombre d'article = 100 000 articles  
 LART = longueur de l'article = 20 octets = 10 mots  
 LCLE = longueur de la clé = 6 octets = 3 mots

Hypothèse soit une taille de poste telle que :

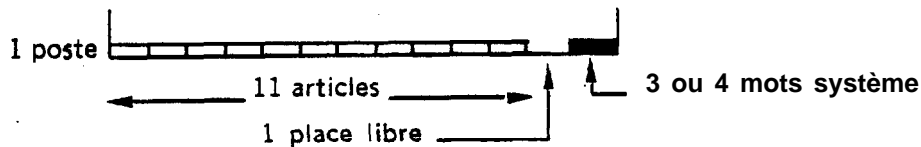
LP = 1 secteur = 256 octets = 128 mots

1°) Calcul du nombre d'articles par poste de niveau information

$$\text{NMAPP} = \text{nombre maximal} = (\text{LP} - (6 \text{ ou } 8)) / \text{LART} - 1$$

- 1 pour la place libre en fin de poste
- 6 si table directe sans homonymes ou table inverse
- 8 si table directe avec homonymes

Ex :  $\text{NMAPP} = (256 - 6) / 20 - 1 = 11$  articles (résidu 5 mots)



soit NAPP = nombre moyen compte tenu d'un taux de remplissage théorique moyen de 75 %.

$$\text{NAPP} @ \text{NMAPP} \cdot \text{TAUX}$$

soit NAPP ~ 8 articles

2°) Calcul du nombre d'index par poste de table d'index

$$\text{NMIPP} = \text{nombre maximal} = ((\text{LP} - 6) / (\text{LCLE} + 2)) - 1$$

Ex. :  $\text{NMIPP} = (256 - 6) / (6 + 2) - 1 = 30$  (résidu 1 mot)

soit NIPP = nombre moyen compte tenu d'un taux de remplissage de 75 %

$$\text{NIPP} = \text{NMIPP} \cdot \text{TAUX}$$

soit NIPP = 22 index



### 3°) Calcul de l'occupation du fichier

NP0 = nb postes niveau 0 =  $N/NAPP = 100\ 000/8 = 12\ 500$  postes  
NP1 = nb postes niveau 1 =  $NPO/NIPP = 12\ 500/22 = 570$  postes  
NP2 = nb postes niveau 2 =  $NPI/NIPP = 570/22 = 26$  postes  
NP3 = nb postes niveau 3 =  $NP2/NIPP$  or  $26 < 30 = 1$  poste

Nombre total de poste = 13 077 postes

Soit 13 077 postes => 26 154 secteurs

Soit environ 103 granules de 32K mots

ou 28 granules de 960 secteurs, etc...

Organisation standard

Organisation grand disque

### 4°) Puissance théorique d'un tel fichier

2 niveaux =  $8 \times 30 \Rightarrow 240$

3 niveaux =  $8 \times 22 \times 30 \Rightarrow 5\ 280$  articles

4 niveaux =  $8 \times 22^2 \times 30 \Rightarrow 116\ 160$  articles

## ANNEXE

<b>A : SYNOPTIQUE</b>	<b>A-1</b>
Les comptes-rendus de FMS	A-1
Correspondance primitive compte-rendu	A-2
FCB	A-4
Valeurs limitées de FMS - FUP	A-11
Taille des fichiers	A-15
Le partage des fichiers	A-16
Schémas de FMS	A-20
Les zones de travail de FMS	A-25
La FAU	A-26
Le WCB	A-27
Le DF	A-29
La DFU	A-30
Le sémaphore de FAU Publique	A-32
Tableau des méthodes d'Accès	A-33

Les Compte-Rendus de FMS	0	Primitive correctement exécutée	(Voir détails dans le MR page 3-7)
	0 £ PR £ '3FFE	Primitive correctement exécutée : READ, WRITE, SKIPB, SKIPF.	à
Avertissements	'6001	Fin d'Article	
	'6002	Début d'Article	
	'6003	Article du fichier plus long	
	'6004	Article du fichier plus court	
	'6005	Article incorrect	
	'6006	Fin de chaîne	
	'6007	Début de chaîne	
Erreurs	'600A	FAU inexistante	
	'600B	FAU existante	
	'600C	Fichier inexistant	
	'6000	Fichier existant	
	'600E	Article inexistant	
	'600F	Article existant	
	'6014	Protection écriture	
	'6015	Permanent de nature différente	
	'6016	Fichier saturé	
	'6017	Fichier trop long	
	'6018	Incompatibilité Primitive Fichier	
	'601A	Erreur d'enchaînement	
	'601E	Fichier occupé	
	'601F	Primitive en cours	
Erreurs filtrées par les super- viseurs RBOS/D RTES/D	'6020	Zone de pavés saturés	
	'6021	FU saturée	
	'6022	Table des fichiers saturés	
	'6028	Erreur de Syntaxe	
	'6029	Adresse de FCB invalide	
	'602A	SU ou FU non gérée par FMS ou IOCS	
	'602B	Méthode d'Accès non gérée	
Erreurs graves	'6032	} Informations Système Invalides	
	'6033		
	'6034		
	'6035	FU verrouillée par IOCS	
	'4000	Erreur hardware : '4000 mot d'état PU	

CORRESPONDANCE PRIMITIVE COMPTE-RENDU

Compte-rendu Primitive \	0	LBU	'1	'2	'3	'4	'5	'6	'7	'A	B	'C	'D	'E	'F	'14	'15	'16	'17
OPEN NEW	X										X		X						X
OPEN OLD	X										X	X				X	X		
CLOSE	X									X									
CREAT	X										X		X						X
CATAL	X									X			X						
DELET	X									X						X			
RENUM	X									X	X								
ALTER	X									X						X			
RENAM	X									X			X			X			
EOJ	X																		
READ	X	X	X							X									
WRITE	X	X	X							X						X			X
SKIPB	X	X		X						X									
SKIPF	X	X	X							X									
REWIND	X									X									
SKEOA	X									X									
IREAD	X				X	X				X				X					
IWRITE	X									X					X	X		X	X
ISUP	X									X				X		X			
IRNAM	X									X					X	X			
IRWRITE	X				X	X				X				X		X			
IRTIX	X				X	X				X									
DREAD	X				X	X				X				X					
DWRITE	X				X	X				X				X		X			
DCRE	X				X	X				X				X		X		X	
DSUP	X									X				X		X			
SIREAD	X				X	X				X				X					
SIRIS	X		X	X	X	X		X	X	X									
SIWRIT	X				X	X	X			X						X			
SIADD	X				X	X				X					X	X		X	
SISUP	X						X			X						X			
SCADD	X									X				X		X		X	
SCREAD	X							X		X				X					
SCWRIT	X									X						X			
SCDREAD	X									X				X					

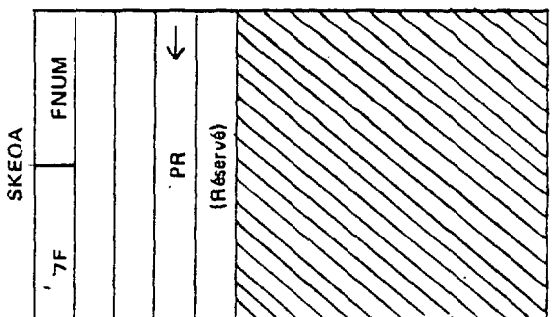
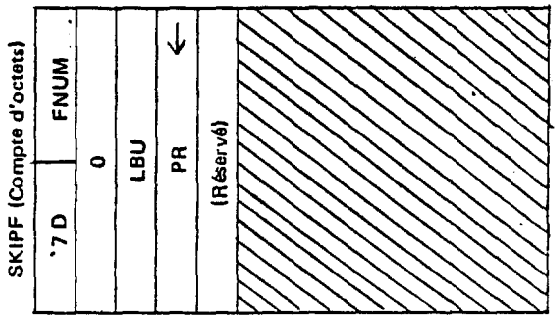
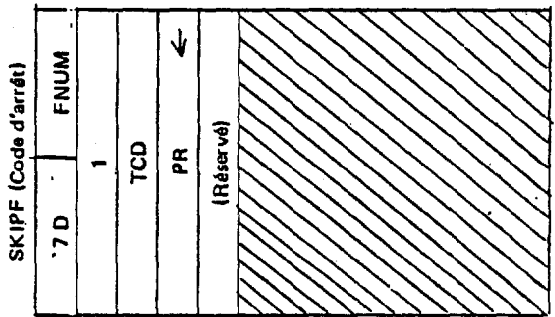
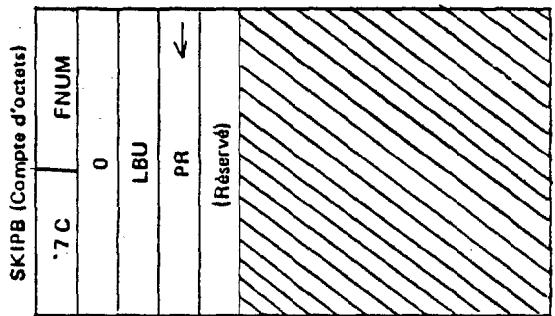
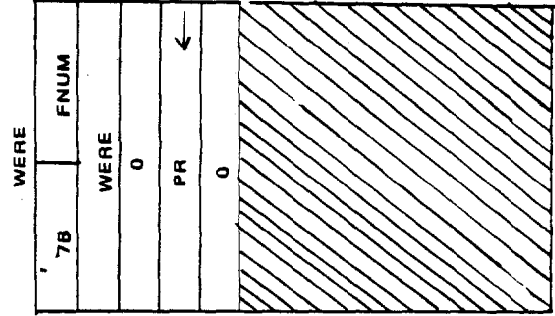
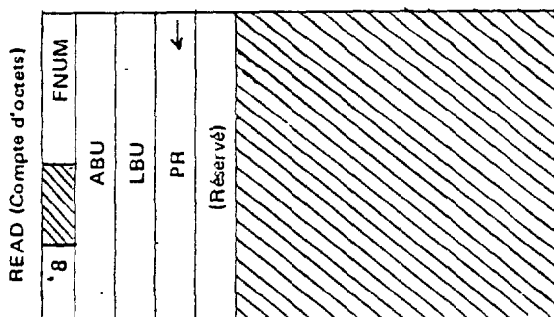
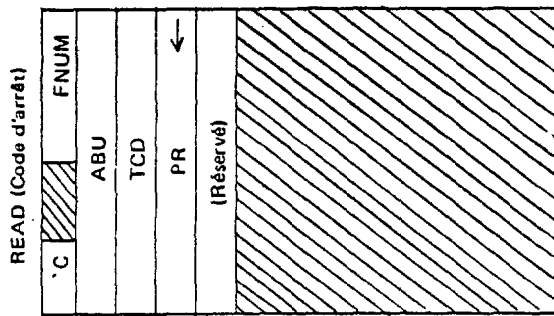
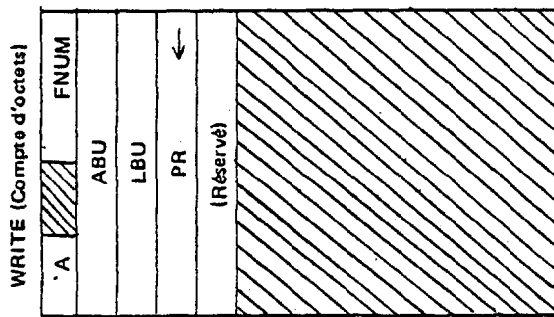
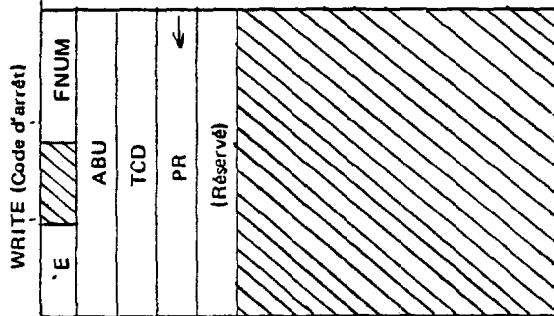
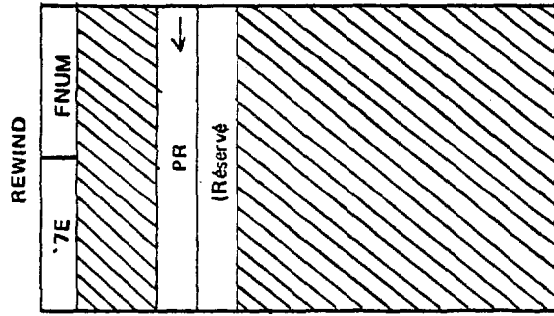
CORRESPONDANCE PRIMITIVE COMPTE-RENDU

Compte-rendu Primitive	'18	'IA	'IE	'IF	'20	'21	'22	'28	'29	'2A	'2B	'32	'33	'34	'35	'36	'37	hard
OPEN NEW				X	X	X		X	X	X	X	X	X	X	X	X	X	X
OPEN OLD		X	X	X	X			X	X	X	X	X		X	X	X	X	X
CLOSE				X				X	X		X	X	X	X	X	X	X	X
CREAT				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
CATAL	X			X			X	X	X		X	X		X	X	X	X	X
DELET			X	X				X	X		X	X	X	X	X	X	X	X
RENUM				X				X	X		X					X	X	
ALTER	X	X	X	X				X	X		X			X	X	X	X	X
RENAM	X		X	X				X	X		X	X		X	X	X	X	X
EOJ								X	X		X	X	X	X	X	X	X	X
READ		X		X				X	X		X	X		X	X	X	X	X
WRITE		X		X		X		X	X		X	X	X	X	X	X	X	X
SKIPB		X		X				X	X		X	X		X	X	X	X	X
SKIPF		X		X				X	X		X	X		X	X	X	X	X
REWIND		X		X				X	X		X	X		X	X	X	X	X
SKEOA		X		X				X	X		X	X		X	X	X	X	X
IREAD	X			X				X	X		X	X		X	X	X	X	X
IWRITE	X			X		X		X	X		X	X		X	X	X	X	X
ISUP	X			X				X	X		X	X		X	X	X	X	X
IRNAM	X	X		X				X	X		X	X		X	X	X	X	X
IRWRITE	X			X				X	X		X	X		X	X	X	X	X
IRTIX	X			X				X	X		X	X		X	X	X	X	X
DREAD	X			X				X	X		X	X		X	X	X	X	X
DWRITE	X			X				X	X		X	X		X	X	X	X	X
DCRE	X			X				X	X		X	X		X	X	X		X
DSUP	X			X				X	X		X	X		X	X	X		X
SIREAD	X			X				X	X		X	X		X	X	X		X
SIRIS	X	X		X				X	X		X	X		X	X	X		X
SIWRIT	X	X		X				X	X		X	X		X	X	X		X
SIADD	X			X				X	X		X	X		X	X	X		X
SISUP	X	X		X				X	X		X	X		X	X	X		X
SCADD	X	X		X				X	X		X	X		X	X	X		X
SCREAD	X			X				X	X		X	X		X	X	X		X
SCWRIT	X	X		X				X	X		X	X		X	X	X		X
SCDREAD	X			X				X	X		X	X		X	X	X		X

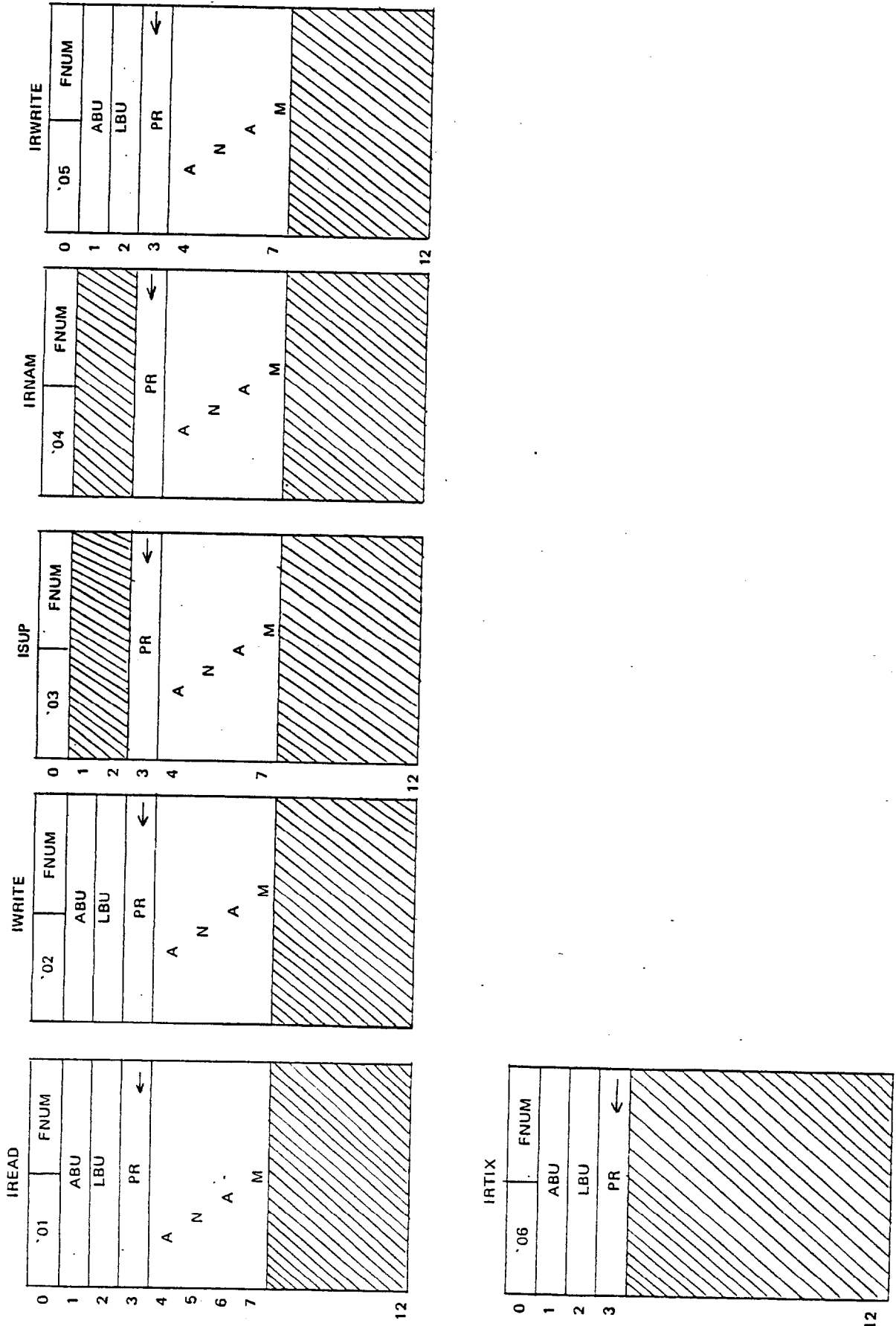




Niveau portion d'article



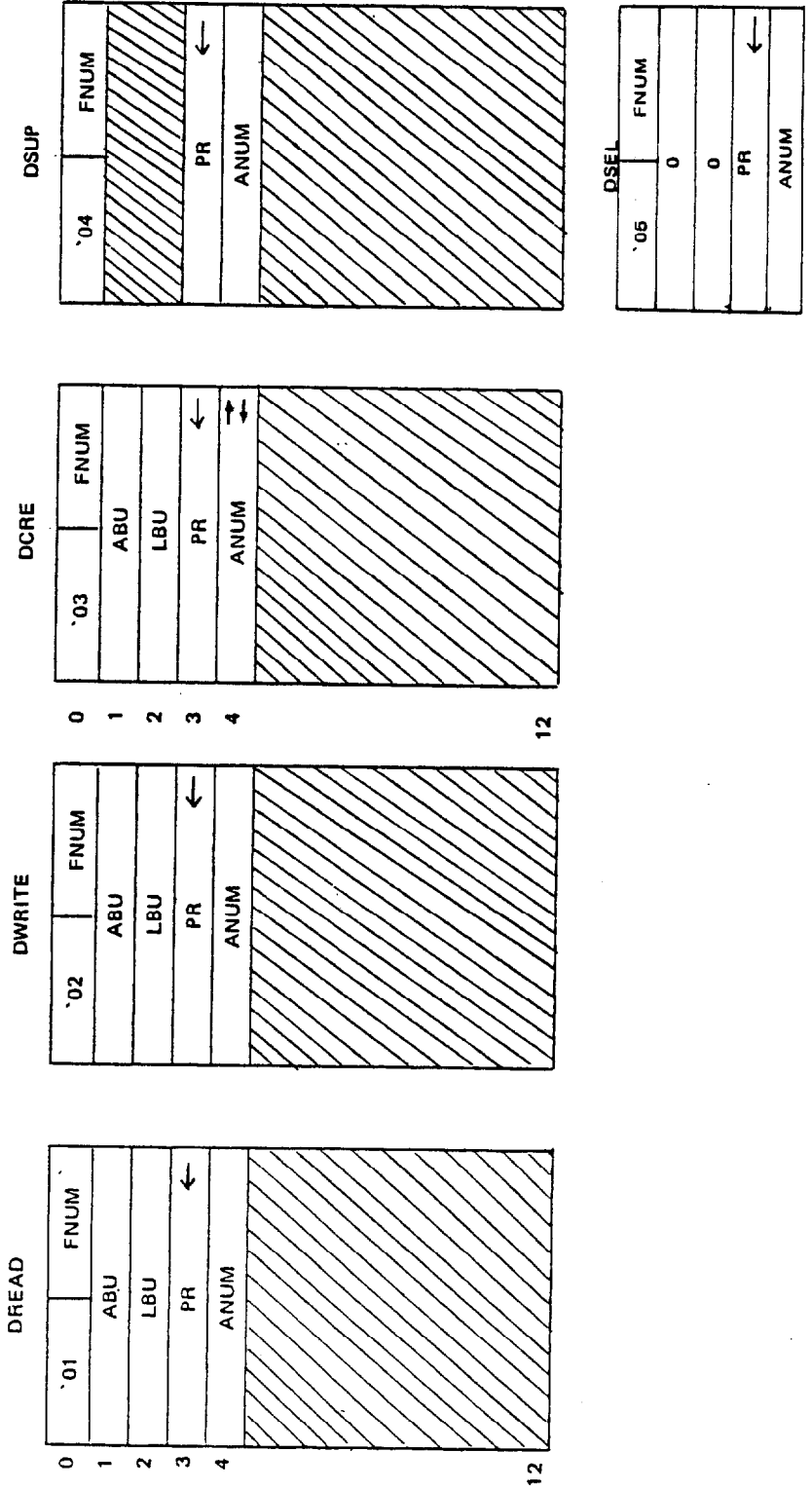
Niveau article d'un fichier Indexé



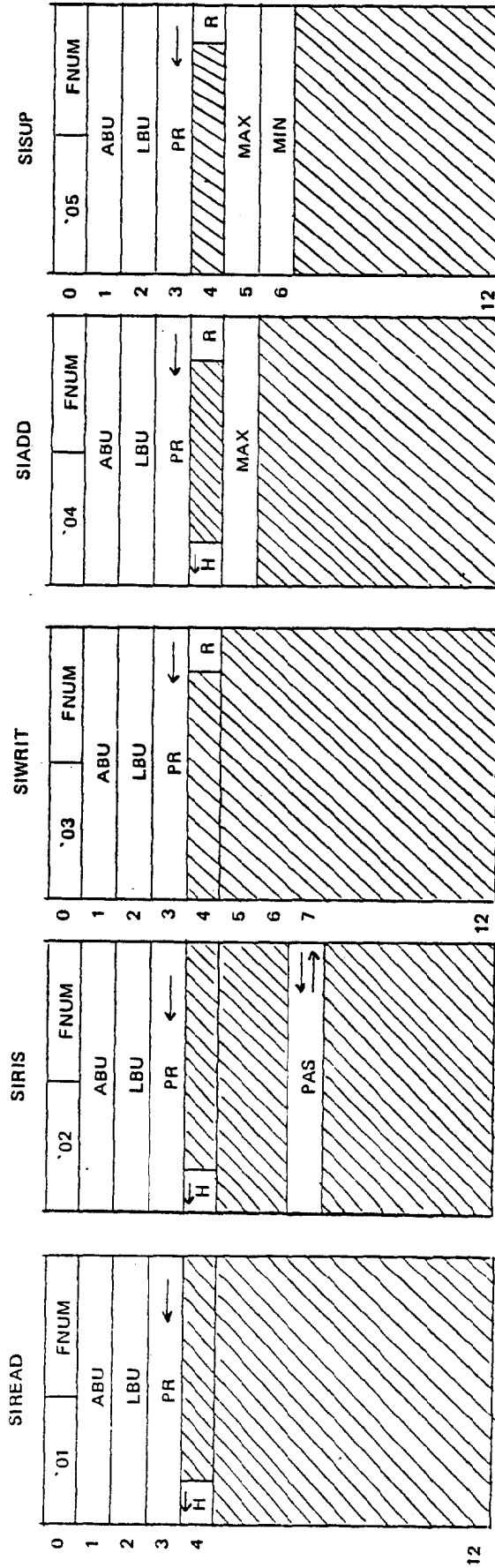
Niveau article de fichiers Direct et Direct à Trous

Direct à Trous (FMS-E)

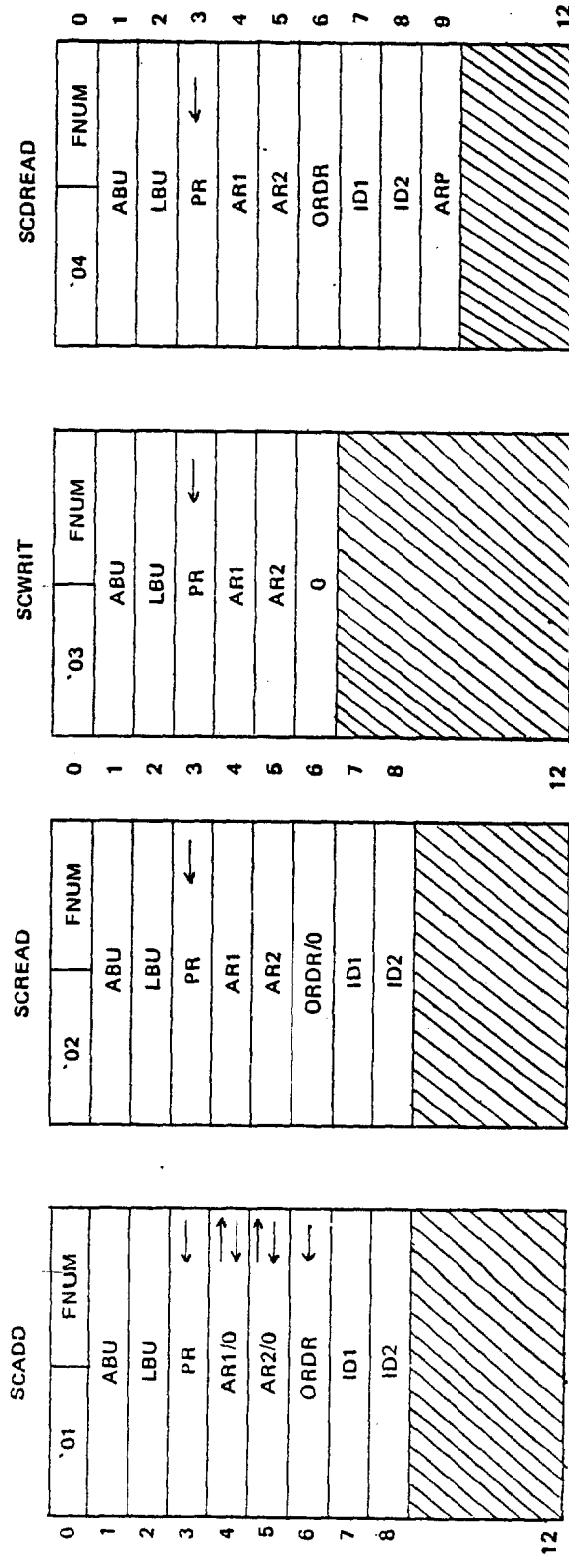
Direct



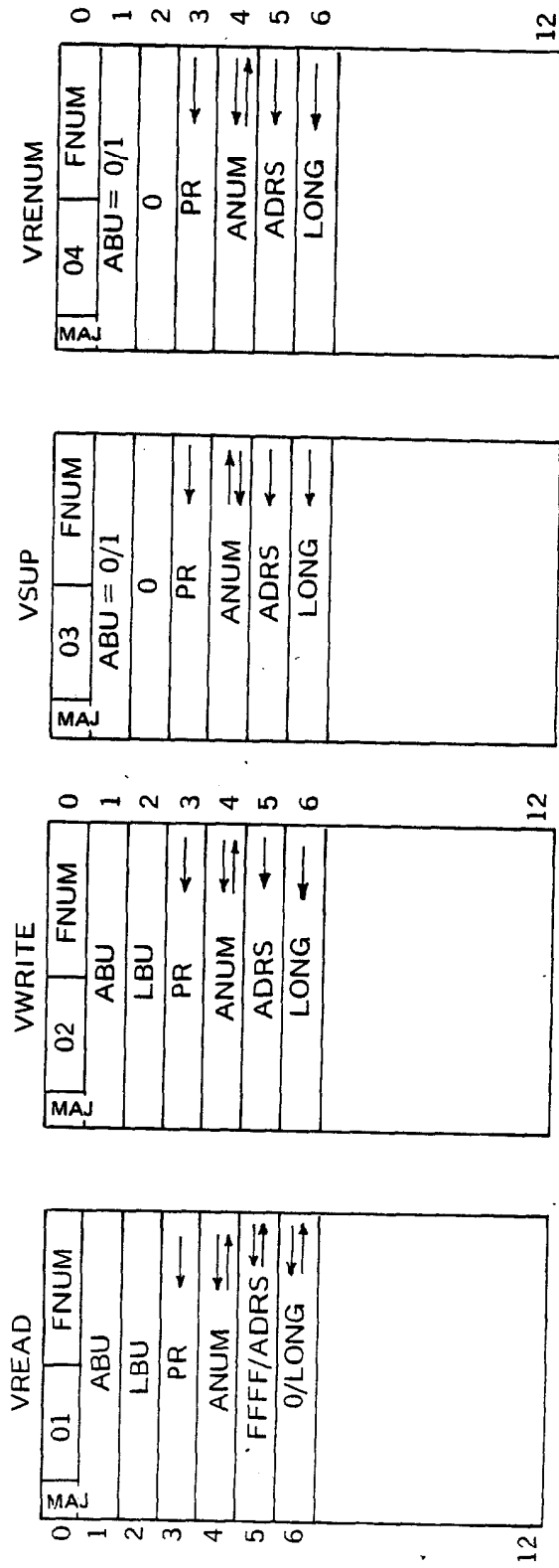
Niveau article d'un fichier Séquentiel Indexé (FMS-E)



Niveau article d'un fichier Séquentiel Chainé (FMS-E)



Niveau article d'un fichier Direct Variable (FMS-E)



## Valeurs limites liées à la gestion des disques par le noyau FMS (KERNEL)

	Taille minimum	Taille maximum
E/S IOCS	1 secteur Remarque : R/W du 1er mot d'un secteur, avec en écriture RAZ de la fin du secteur	Limite IOCS : 8 K mots - 1 ou 6 K sur disque cartouche 5 M. oct.
E/S Article	0 ou 1 mot	32 K mots - 1 si TG < limite IOCS FMS-E limite IOCS si TG > limite IOCS FMS-G 32 K mots - 1
Portion E/S d'article	0 ou 1 mot	FMS-E limite IOCS FMS-G 8 K mots - 1
Buffer travail	1 secteur	voir méthode d'accès limite IOCS ou 32 K mots - 1
E/S demandée à IOCS par FMS	1 secteur OU le 1er mot d'un secteur	- ou demande utilisateur buffer de travail INF - Taille utile du granule - limite IOCS
	3 secteurs	FMS-E 192 secteurs FMS-G ~ 400 K mots
	- occupation 1 secteur - LTAG = 1 mot chaîne de bit - FMS-E + 3 mots système - FMS-G + 7 mots système	FMS-E 1 secteur LTAG = 125 mots FMS-G 2048 mots LTAG = 2041 mots

## Valeurs limites liées à la gestion des disques par le noyau FMS (KERNEL)

	Taille minimum	Taille maximum	Nombre minimum	Nombre maximum
FU IOCS	1 cylindre	NBCYL Nombre total de cylindres du disque	D1 2 D2	<sup>29</sup> D1 à DF, E1 à EB, ED, EF
Secteur	LSEC = 128 mots	FMS-E 128 mots FMS-G 2048 mots		FMS-E 32 767 par FU FMS-G 2 <sup>31</sup> / LSEC
FU FMS	FU IOCS	FMS-E } NBCYL - 1 4 méga-mots FMS-G } NBCYL 2 milliards de mot	1 D2	28 D2 à DF, E1 à EB, ED, EF
Granule	3 secteurs	FMS-E 256 Sect. FMS-G 32 K - 1 Sect.	FMS - E 2 FMS - G 1	FMS-E 2000 FMS-G 32655
Fichiers	Logiquement : 0 mot Physiquement : 1 granule	Log. voir Méth. d'accès FMS-E : 4 méga-mots FMS-G : 2 milliards de mots Phys. 128 granules pour les fichiers statiques	1 par FU (ou espace disque)	FMS-E : 1999 FMS-G : 32656



Valeurs limites des fichiers dues au logiciel FMS-E

Méthodes d'accès	3 chiffres dans l'ordre Min, Raisonnable, Max				
	NOL	Statique ou Dynamique	Taille d'article	Nombre d'articles	Taille fichier
0	Séquentiel	Dyn.	Portion d'article - 0 mot - n. secteurs - 8 K mots ou limite IOCS si TG lui est supérieur	1 " "	0 mots illimité "
1	Indexé	Dyn.	1 mot 6 K mots 64 K mots	0 600 8159	Long. TIX 100 K mots 2 M mots + LTX
2	Direct	Stat.	1 mot 32 K mots "	1 65 535 "	1 mot 4 méga-mots "
2	Direct à trous	Stat.	Idem DIR	Idem DIR	Idem DIR
3	Séquentiel indexé	Stat.	1 mot 100 mots 2 K mots	1 100 000 200 000 ~	3 secteurs 4 méga-mots 32 767 postes de 128 mots 128 mots
4	Séquentiel chaîné	Stat.	1 mot 10 mots 12 K mots	Nombre de chaînes 1 20 000 32 767	2 secteurs 2 M. mots (32 767 postes de 64 mots, ou 344 postes de 12 K mots)
5	Direct Longueur variable		1 mot 6 K mots 32 K mots	0 1000 16 384 (LIX = 4)	Long TIX 200 K mots 32 K sect. + LTX

## Valeurs limites des fichiers dues au Logiciel

Méthodes d'accès	3 chiffres dans l'ordre Min, Raisonnable, Max		
	NOL Statique ou Dynamique	Taille d'articles	Nombre d'articles
0 Séquentiel  Dyn.	Portion d'article 0 mot 4 secteurs 8 K mots	1 " "	0 mot illimité "
1 Indexé  Stat.	1 mot 6 K mots 64 K mots	0 600 8159	Long. TIX 100 K mots 2 M mots LTX
2 Direct  Stat.	1 mot 32 K mots "	1 65 535 "	1 mot 2 milliards mots "
2 Direct à trous  Stat.	Idem DIR	1 2 milliards "	Idem DIR
3 Séquentiel indexé  Stat.	1 mot 100 mots 2 K mots	1 100 000 200 000	3 secteurs 145 M mots (65 535 postes de 6 K mots)
4 Séquentiel chaîné  Stat.	1 mot 10 mots 12 K mots	Nombre de chaînes 1 20 000 32 767	2 secteurs 2 M mots (32 767 postes de 64 mots, ou 344 postes de 12 K mots)
5 Direct Longueur variable	1 mot 6 K mots 32 K mots	0 1 000 16 384	Long. TIX 200 K mots 32 K sect. + LTIX

## Taille des fichiers en nombre de secteurs

NBG	1	2	4	5	8	10	16	20	32	50	64	100	128	200	256	500	512	1 000	1 024
TG	7	14	28	35	56	70	112	140	224	350	448	700	896	1 400	1 792	3 500	3 581	7 000	7 168
	15	30	60	75	120	150	240	300	480	750	960	1 500	1 920	3 000	3 840	7 500	7 680	15 000	15 360
	23	46	92	115	184	230	368	460	736	1 150	1 472	2 300	2 944	4 600	5 888	11 500	11 776	23 000	23 552
	31	62	124	155	248	310	496	620	992	1 550	1 984	3 100	3 968	6 200	7 936	15 500	15 872	31 000	31 744
	47	94	188	235	376	470	752	940	1 504	2 350	3 008	4 700	6 016	9 400	12 032	23 500	24 064	47 000	48 128
	63	126	252	315	504	630	1 008	1 260	2 016	3 150	4 032	6 300	8 064	12 600	16 128	31 500	32 256	63 000	64 512
	95	190	380	475	760	950	1 520	1 900	3 040	4 750	6 080	9 500	12 160	19 000	24 320	47 500	48 640	95 000	97 280
	127	254	508	635	1 016	1 270	2 032	2 540	4 064	6 350	8 128	12 700	16 256	25 400	32 512	63 500	65 024	127 000	130 048
	191	382	764	955	1 528	1 910	3 056	3 820	6 112	9 550	12 224	19 100	24 448	38 200	48 896	95 500	97 792	191 000	195 584
	255	510	1 020	1 275	2 040	2 550	4 080	5 100	8 160	12 750	16 320	25 500	32 640	51 000	65 280	127 500	130 560	255 000	261 120

Limite de 128 granules (fichiers statistiques)

Limite de 399 cylindres de 6 K mots

Limite de 4 M mots

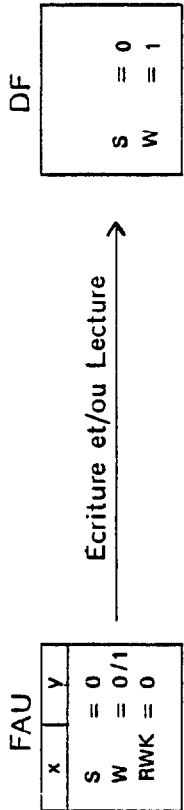
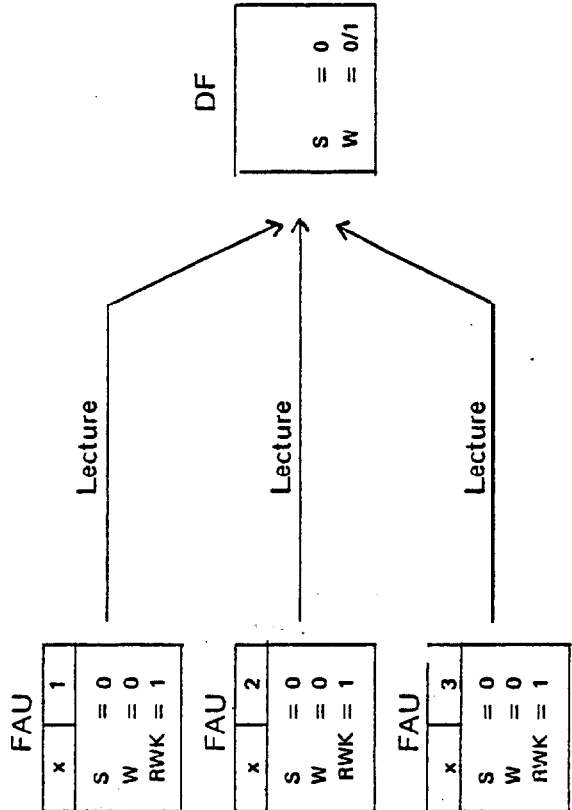
Le Partage des Fichiers

Type	Temporaire	Permanent Simultané (S = 1)																																
Les usagers	N Usagers	N Usagers																																
Le partage	NON partageable	Partage en Lecture																																
	<p>Mono-accès</p> <p>OPEN NEW</p>																																	
	<p>Usager x</p> <p>FAU</p> <table border="1" style="margin-left: 40px;"> <tr><td>USR</td><td>FNUM</td></tr> <tr><td> </td><td> </td></tr> </table> <p>DF</p> <table border="1" style="margin-left: 40px;"> <tr><td>A</td><td>B</td></tr> <tr><td>C</td><td>D</td></tr> <tr><td>E</td><td>F</td></tr> <tr><td>1</td><td>USR</td></tr> </table> <p>FNAN</p> <p>"PUBW"</p> <p>Écriture - Lecture</p>	USR	FNUM			A	B	C	D	E	F	1	USR	<p>Usager 1</p> <p>Usager 2</p> <p>Usager 3</p> <p>FAU</p> <table border="1" style="margin-left: 40px;"> <tr><td>1</td><td>1</td></tr> <tr><td>S = 1</td><td>W = 0</td></tr> <tr><td>RWK = 0</td><td> </td></tr> </table> <p>FAU</p> <table border="1" style="margin-left: 40px;"> <tr><td>2</td><td>1</td></tr> <tr><td>S = 1</td><td>W = 0</td></tr> <tr><td>RWK = 0</td><td> </td></tr> </table> <p>FAU</p> <table border="1" style="margin-left: 40px;"> <tr><td>3</td><td>1</td></tr> <tr><td>S = 1</td><td>W = 0</td></tr> <tr><td>RWK = 0</td><td> </td></tr> </table> <p>DF</p> <table border="1" style="margin-left: 40px;"> <tr><td>S = 1</td><td>W = 0/1</td></tr> </table> <p>Lecture</p> <p>Lecture</p> <p>Lecture</p>	1	1	S = 1	W = 0	RWK = 0		2	1	S = 1	W = 0	RWK = 0		3	1	S = 1	W = 0	RWK = 0		S = 1	W = 0/1
USR	FNUM																																	
A	B																																	
C	D																																	
E	F																																	
1	USR																																	
1	1																																	
S = 1	W = 0																																	
RWK = 0																																		
2	1																																	
S = 1	W = 0																																	
RWK = 0																																		
3	1																																	
S = 1	W = 0																																	
RWK = 0																																		
S = 1	W = 0/1																																	

Le Partage des Fichiers

Les usagers	Le partage	Les Unités d'Accès aux Fichiers	Type																												
N Usagers	1 seul usager en Écriture OPEN OLD	<p>FAU</p> <table border="1" data-bbox="397 1220 537 1390"> <tr><td>x</td><td>y</td></tr> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 1</td></tr> <tr><td>RWK</td><td>= 0</td></tr> </table> <p>Usager x</p> <p>Écriture — Lecture</p> <p>DF</p> <table border="1" data-bbox="397 585 537 755"> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 1</td></tr> </table>	x	y	S	= 1	W	= 1	RWK	= 0	S	= 1	W	= 1	Permanent Simultané (S = 1)																
x	y																														
S	= 1																														
W	= 1																														
RWK	= 0																														
S	= 1																														
W	= 1																														
1 Usager	Multi-accès en Lecture OPEN OLD	<p>FAU 1</p> <table border="1" data-bbox="625 1220 764 1390"> <tr><td>x</td><td>1</td></tr> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 0</td></tr> <tr><td>RWK</td><td>= 0</td></tr> </table> <p>Usager x</p> <p>Lecture</p> <p>FAU 2</p> <table border="1" data-bbox="820 1220 959 1390"> <tr><td>x</td><td>2</td></tr> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 0</td></tr> <tr><td>RWK</td><td>= 0</td></tr> </table> <p>Usager x</p> <p>Lecture</p> <p>FAU 3</p> <table border="1" data-bbox="1008 1220 1148 1390"> <tr><td>x</td><td>3</td></tr> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 0</td></tr> <tr><td>RWK</td><td>= 0</td></tr> </table> <p>Usager x</p> <p>Lecture</p> <p>DF</p> <table border="1" data-bbox="820 585 959 755"> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 0/1</td></tr> </table>	x	1	S	= 1	W	= 0	RWK	= 0	x	2	S	= 1	W	= 0	RWK	= 0	x	3	S	= 1	W	= 0	RWK	= 0	S	= 1	W	= 0/1	
x	1																														
S	= 1																														
W	= 0																														
RWK	= 0																														
x	2																														
S	= 1																														
W	= 0																														
RWK	= 0																														
x	3																														
S	= 1																														
W	= 0																														
RWK	= 0																														
S	= 1																														
W	= 0/1																														
	Mono-accès en Écriture OPEN OLD	<p>FAU</p> <table border="1" data-bbox="1258 1220 1398 1390"> <tr><td>x</td><td>y</td></tr> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 1</td></tr> <tr><td>RWK</td><td>= 0</td></tr> </table> <p>Usager x</p> <p>Écriture — Lecture</p> <p>DF</p> <table border="1" data-bbox="1258 585 1398 755"> <tr><td>S</td><td>= 1</td></tr> <tr><td>W</td><td>= 1</td></tr> </table>	x	y	S	= 1	W	= 1	RWK	= 0	S	= 1	W	= 1																	
x	y																														
S	= 1																														
W	= 1																														
RWK	= 0																														
S	= 1																														
W	= 1																														

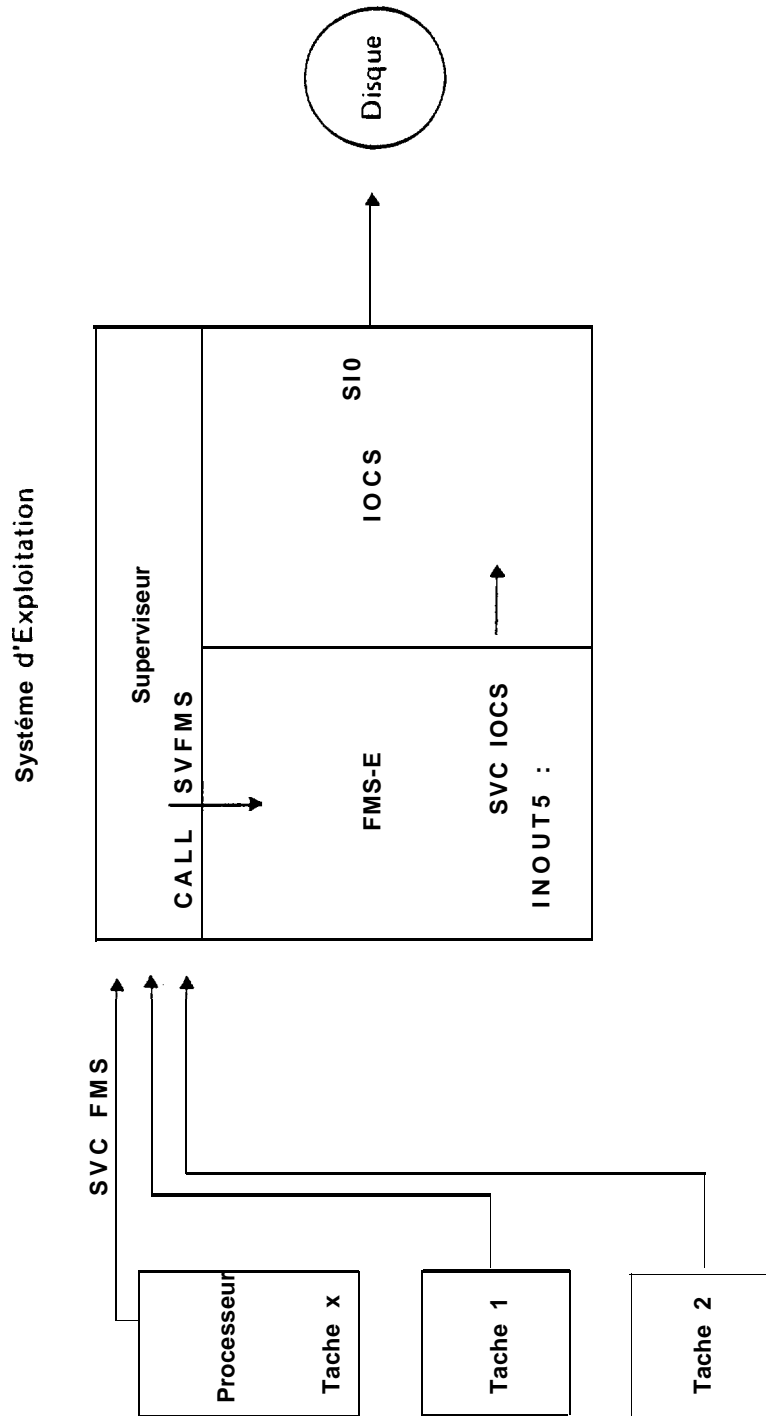
La Partage des Fichiers

		Type
		Permanent NON Simultané (S = 0)
Les Unités d'Accès aux Fichiers		
Voir c-dessous		
Le partage	1 seul Usager à la fois	
Les Usagers	N Usagers	
	Mono-accès RWK = 0	 <p>FAU</p> <p>x y</p> <p>S = 0 W = 0/1 RWK = 0</p> <p>Usager x</p> <p>Ecriture et/ou Lecture</p> <p>DF</p> <p>S = 0 W = 1</p>
	Multi-accès Lecture RWK = 1	 <p>FAU</p> <p>x 1</p> <p>S = 0 W = 0 RWK = 1</p> <p>Usager x</p> <p>Lecture</p> <p>FAU</p> <p>x 2</p> <p>S = 0 W = 0 RWK = 1</p> <p>Usager x</p> <p>Lecture</p> <p>FAU</p> <p>x 3</p> <p>S = 0 W = 0 RWK = 1</p> <p>Usager x</p> <p>Lecture</p> <p>DF</p> <p>S = 0 W = 0/1</p>
	OPEN OLD	

Le Partage des Fichiers

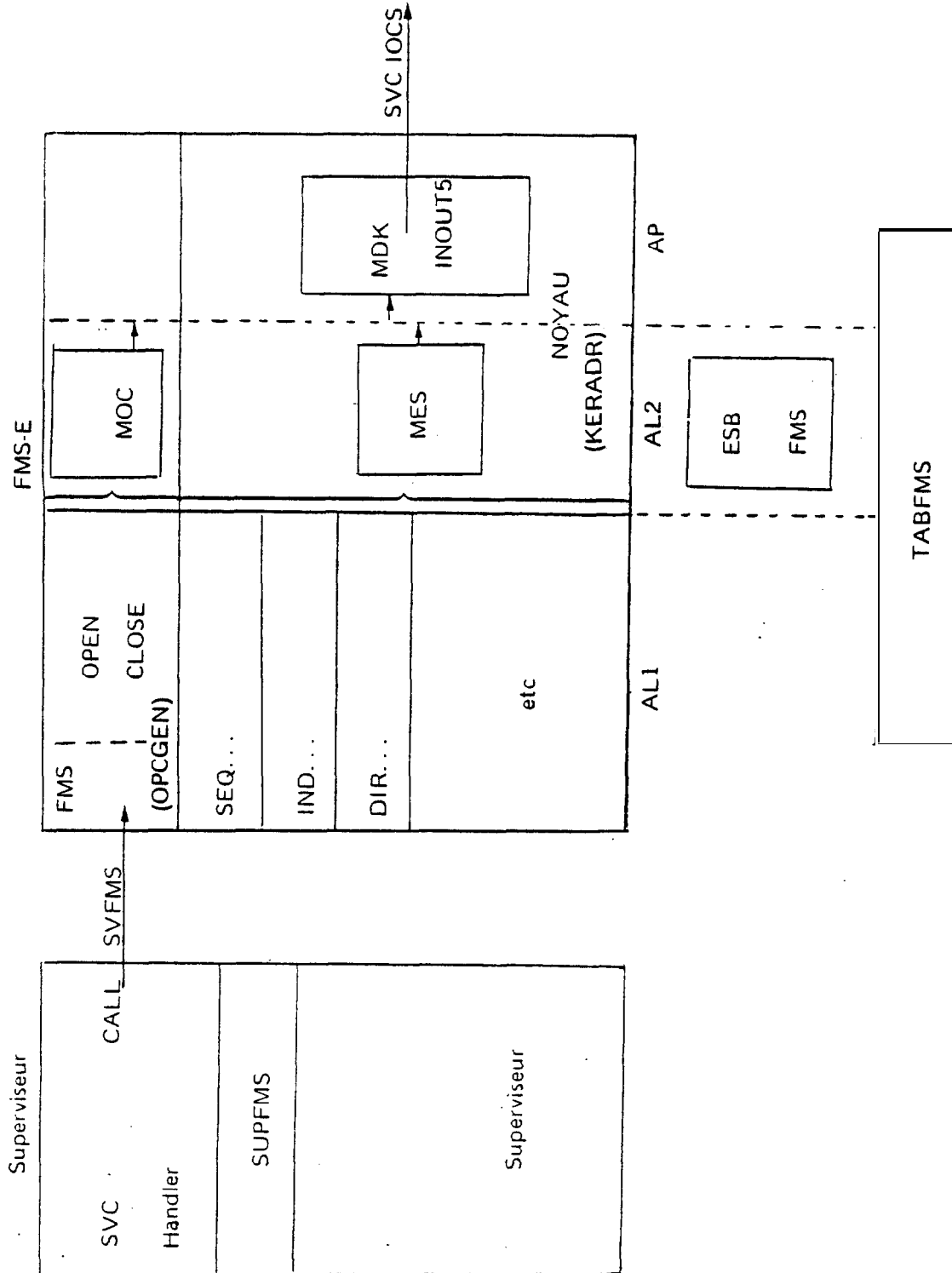
Les Usagers	Le partage	Les Unités d'Accès aux Fichiers	Type
	<p>Pour : Direct et Direct à Trous</p> <p>Multi-accès Lecture</p> <p>= R W K = 1</p> <p>Avec possibilité d'écriture, réservée à la 1ère ouverture dans le temps.</p> <p>OPEN OLD</p>	<p>Les Unités d'Accès aux Fichiers</p> <p>(pour le 1er OPEN OLD)</p> <p>Usager x</p> <p>Usager x</p> <p>Usager x</p>	Permanent NON Simultané (S = 0)
1 Usager	<p>Pour : Direct</p> <p>Multi-accès Ecriture</p> <p>= R W K = 3</p> <p>OPEN OLD</p>	<p>Usager x</p> <p>Usager x</p> <p>Usager x</p>	Permanent NON Simultané (S = 0)

Schéma d'un Système d'Exploitation (2.1.1)

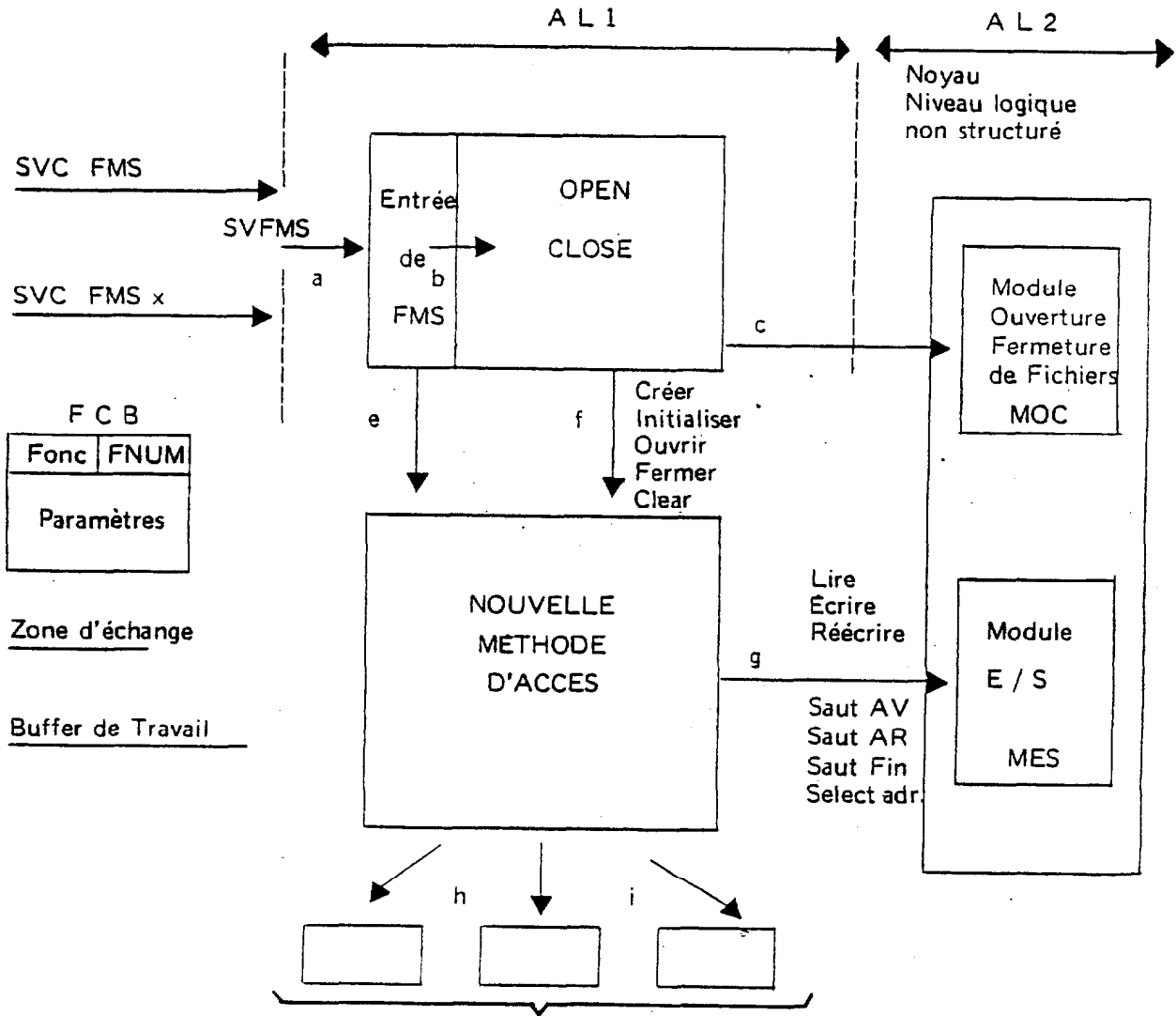




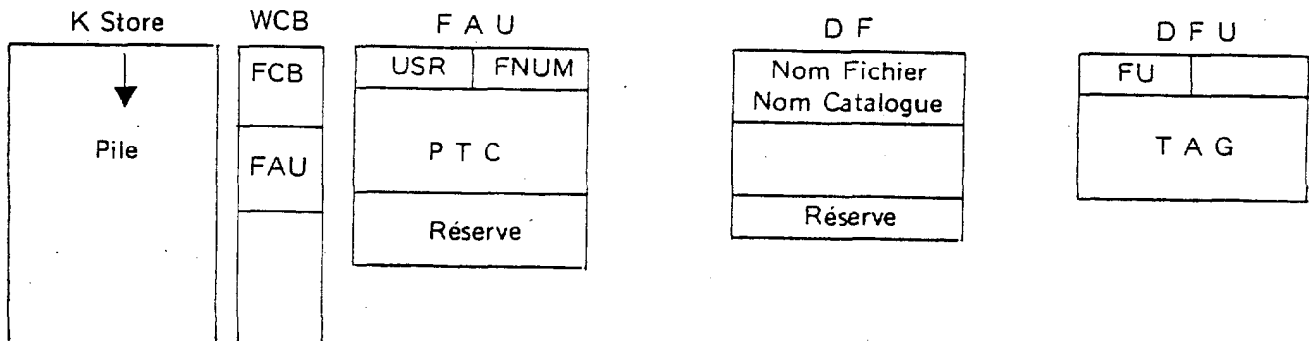
FMS-E et les éléments link-édités (2.1.2)



Environnement d'une Méthode d'Acc7s dans FMS-E (2.2.1)

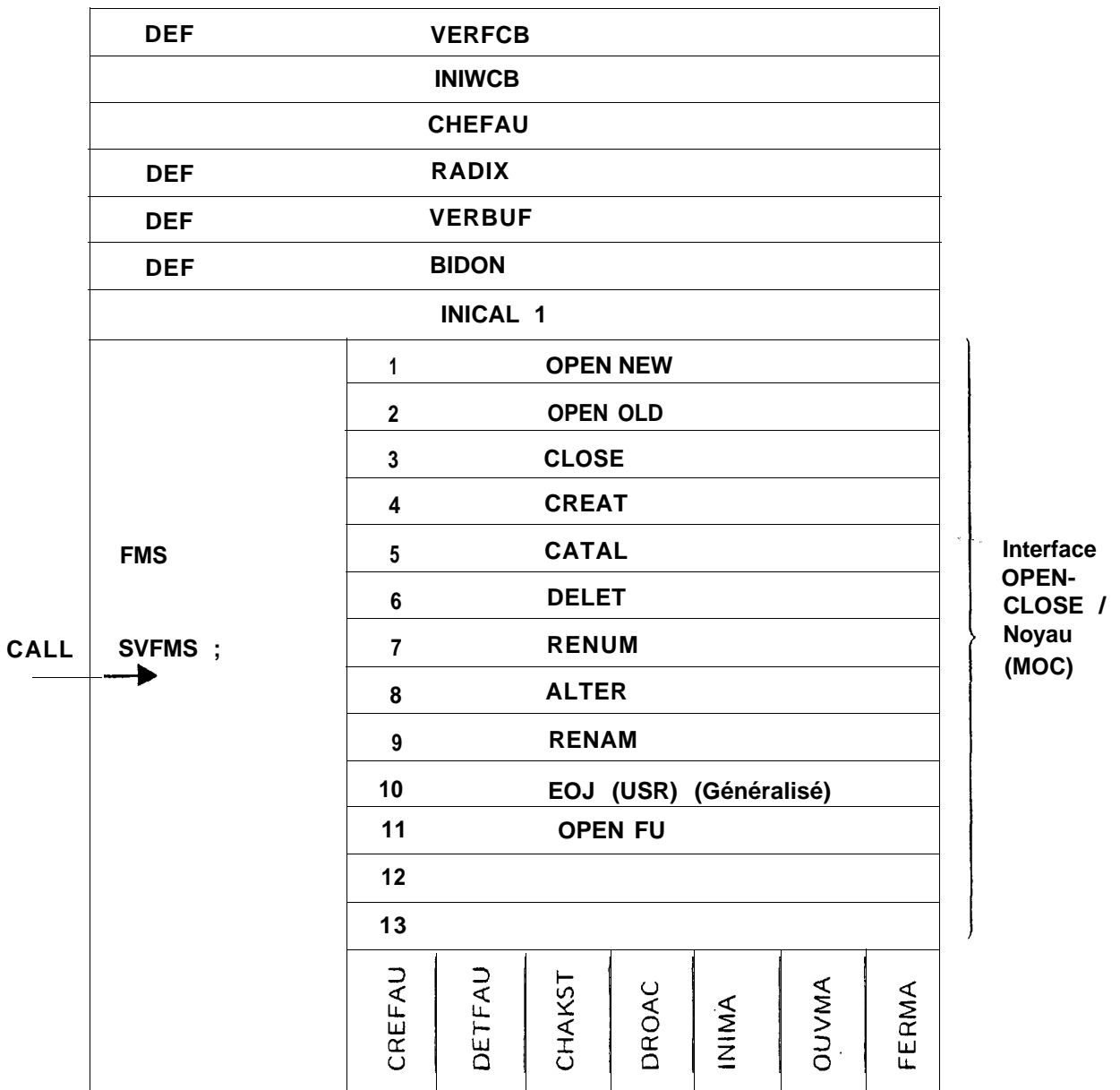


Sous-programmes de service FMS et Superviseur



LE MODULE OPEN-CLOSE (2.4)

Schéma général



Interface FMS /  
Méthode d'Accès

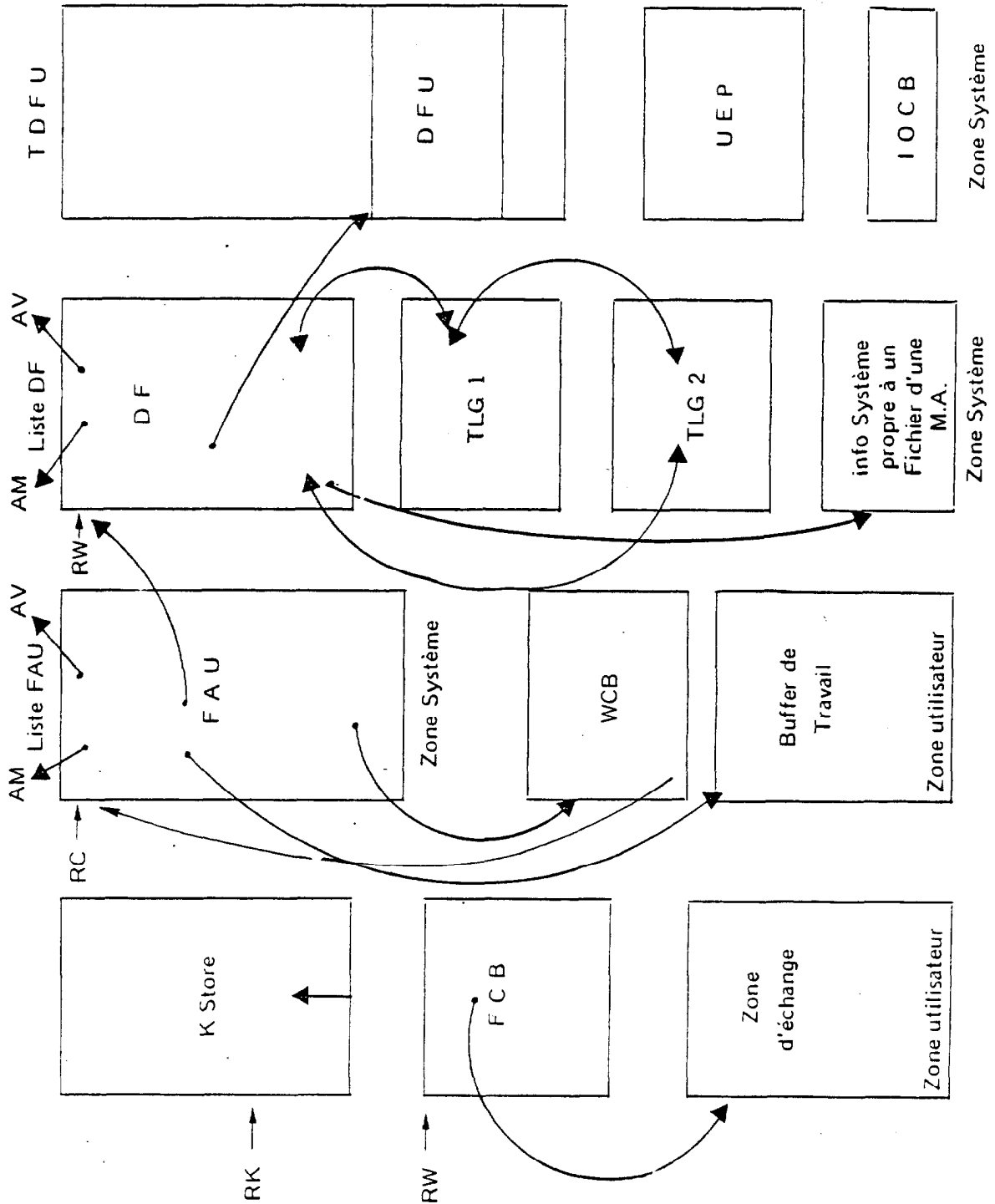
Interface OPEN-CLOSE /  
Méthode d'Accès

LE NOYAU DE FMS-E (2.3)

Schéma général du Noyau KER ADR



Les zones de travail de FMS-E





Le WCB résumé

0	WSLOBU	} Recopie du FCB
1	WSREBU	
2	WFONFNU	
3	WABU	
4	WLBU	
5	WPR	
6	WFNAM1	
7	WFNAM2	
8	WFNAM3	
9	WPUBW	
10	WFTYSFU	
11	WLART	
12	WTART	
13	WNART	
14	WLCLE	
15	WABUEP	} Recopie de la FAU
16	WLBUEP	
17	WPBUEP	
18	WLBUFSEC	
19	WCS	
20	WCCC1	
21	WCCC	
22	WAM1	
23	WAM	
24	WAV1	
25	WAV	
26	WPPFTYP	
27	WBOOLA	
28	WEBOOLA	} Recopie de la FAU
29	WRESTA	
30	WDEJA	
31	WBORMAX	
32	WATIX	
33	WLBTIX	
34	WPTIX	
35	WSLOTIX	
36	WSLETIX	
37	WSLOBT	
38	WSLEBT	

Le WCB résumé

39	WLAM 1
40	WLAM
41	WLAV
42	WLAV 1
43	WADR 1
44	WADR 2
45	WAZU
46	WSLOZE
47	WSLEZE
48	WSLOMDK
49	WMDKCO
50	WMDKCI
51	WMDKC 2
52	WATLG
53	WSAVAR 1
54	WSAVAR
55	WAD 1
56	WAD
57	WLGMO
58	WMOECH
59	WFPV
60	WPRE 1
61	WPRE
62	WSLO
63	WSLE
64	WFAU

MEMOIRES DE TRAVAIL





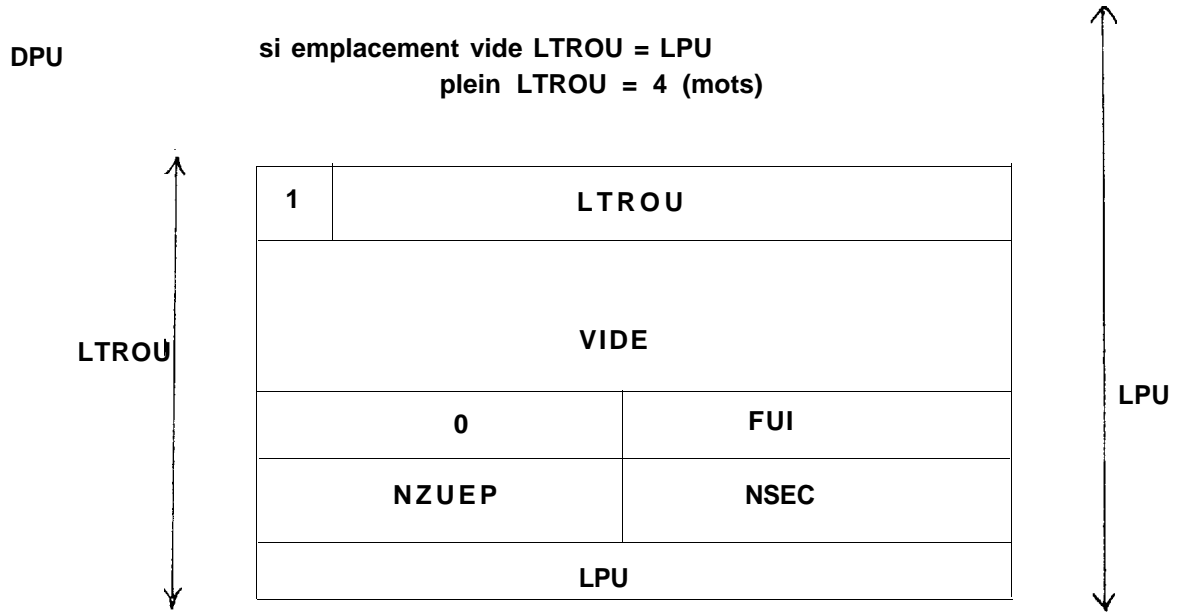
Le DF - Descripteur de fichier

	0	+ , 8	15	
0	Pointeur AVAL - AVALF			
1	Pointeur AMONT - AMONTF			
2	FNAM 1			
3	FNAM 2			
4	PUBW			
5	LTLGD	EMA	SID S W AFI OF	LTGFTYP
6	FS			
7	FF1			
8	FF			
9	DD1			
10	DD			
11	TART'			
12	NART1'			
13	NART'			
14	USRF	CPTFi		USRCPI
15	ADKDF			
16	ADFU			
17	ATLG			
18	AFTLG			
19	LTLG	RWKF	PROGIC AFILUC 105w/ VIDE MOUF	BOOLF
20	LTG			
21	LSEC			
22	NZUEP	LDF		
23	ADRSF			
24	LTIX / APS / NOSUP			
25	ABLIBF			
26	NBLIBF			
27	ADTF			
28	AFTF			
29	ADEG			



TDFU

ENTDFU			
	D2	DFU	<p>1er CAS : Macro-Compatibles en en tête seulement</p> <p>Gestion statique</p> <p>L'emplacement de PU ne pourra jamais être monté car FUI = 0</p>
ENT	D4	DFU	
ENT	D8	DFU	
	1	LTROU DPU                      FUI = 0	
	1	LTROU	<p>2ème CAS : Volume Vide</p> <p>Sera configuré au montage de volume sur FUI</p>
		VIDE	
		D P U                      F U I	
	E1	DFU	<p>3ème CAS : Volume Préinitialisé</p> <p>Sera reconfiguré au montage de volume sur FUI E1 (initiale) est utilisable tel quel pour les FU E1 E2 E3 la taille de la DPU ( LPU) est définie par le Max (PUNBGRAN, SNBGRAN) les FU d'un volume démontable Ne peuvent être définies ENTRY.</p>
	E2	DFU	
	E3	DFU	
	1	LTROU	
		VIDE	
		DPU                      FUI	



OFU = 0 org STD  
OFU = 1 org GDI

0	F U	OFU	PRESE	ETATAE
1	CPTR			
2	LTAGM			
3	NUZUEP	NSEC		
4	TG			
5	NBGL			
6	NBG			
7	LDF			
8	ADTF			
9	AFTF			
10	ADEG			
11	T A G			

généralisé ou recon-  
figuré dynamique-  
ment

A l'ouverture  
de la FU provient  
du disque  
OFU = 0 ou 1

A l'ouverture de la  
FU : provient du  
disque OFU = 1

Sémaphore de FAU publique (voir manuel de Référence 3.5.3)

	0	7	8	15
0	@ F A U			
1	USRP		USR	
2	CPTRA			
3	USRC		USRPRE	
4	NOTACH			
5	SEMFAU			
6	<p>File de bits de SEMFAU</p>			
7				
8				
9				
10				
11				
12				
13				
14	N° de ZBT pour BT			
15	N° ZBT pour TIX (Indexé)			
16				
17				
24				

Tableau général d'identification des Méthodes d'Accès (2.5.1)

FONCTION	Nom de SVC	Numéro de Requête	Numéro de Méthode d'Accès	Nom de Méthode d'Accès	POROPC etc.
Traitement général FMS Traitement OPEN CLOSE	FMS	38	-1	OPC	POROPC etc.
Séquentiel	FMS	39	0	SEQ	
Indexé	FMSI	3A	1	IND	
Direct et Direct à Trous	FMSD	3B	2	DIR DIT	
Séquentiel Indexé	FMSX	28	3	SIX	
Séquentiel Chainé	FMSC	29	4	SCH	
Direct Longueur Variable	FMSV	2A	5	DIV	
		2B	6		
		2C	7		
		2D	8		

Distribution codes/Codes de diffusion			
Customers : Clients :			
Internal : Interne :			

**DELIVERY ADDRESS**  
ÉTIQUETTE ADRESSE

**Bull MTS**

1, Rue de Provence  
B.P. 208  
38432 ÉCHIROLLES CEDEX / FRANCE