

MASTER INFORMATIQUE DEPARTEMENT D'INFORMATIQUE (FSEA)

Année académique 2023-2024

Dr BATOUMA NARKOY Master Informatique : BDA

1

Chapitre 2

Stockage des données Accès aux données

Dr BATOUMA NARKOY Master Informatique : BDA

2

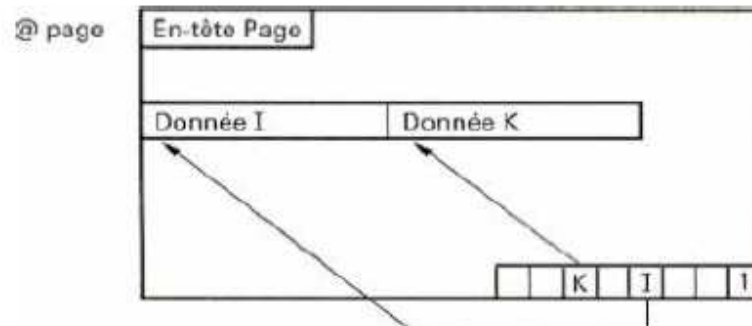
Support de stockage des données

- Notion de page et d'adresse;
- Notion de segment;
- Format de stockage d'un n-uplet;
- Mémoire virtuelle/mémoire tampon

Notion de page et d'adresse

- Les données d'une base de données sont stockées dans des blocs de taille fixe que l'on appelle des **pages**.
- **L'adresse** permanente d'une donnée peut être définie comme l'adresse d'une page sur disque concaténée à la localisation de la donnée dans cette page.
- Afin de permettre de déplacer les données contenues dans une page sans changer leurs adresses, de nombreux systèmes ont recours à une table d'indirection stockée en fin de page (diapo suivante) contenant la position de chaque donnée dans la page.
- Une adresse est alors constituée de l'adresse de la page sur disque et d'un index dans cette page.

Notion de page et d'adresse



L'adresse de la donnée I est : @ page / I
où / représente l'opérateur de concaténation

Structure physique d'une page et notion d'adresse

Dr BATOUMA NARKOY Master Informatique : BDA

5

Notion de segment

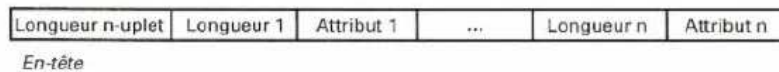
- Chaque page regroupe généralement des données concernant une seule entité logique.
- Entité logique: relation, un index ou toute autre entité regroupant des données ayant des caractéristiques communes.
- Une page contient des données concernant un seul index ou une seule relation.
- La notion de *segment* se définit comme étant l'ensemble de pages regroupant toutes les données d'une entité logique.
- Selon les systèmes, la base de données peut être constituée d'un seul fichier contenant tous les segments, d'un fichier par segment ou de plusieurs fichiers par segment.

Dr BATOUMA NARKOY Master Informatique : BDA

6

Format de stockage d'un n-uplet

- Les n-uplets d'une même relation sont tous stockés dans un même format selon lequel les attributs sont ordonnés.
- Le dictionnaire de données (méta-base) permet de faire la correspondance entre les noms des attributs et la position relative de ces attributs dans le format de stockage de la relation correspondante.
- La structure interne des n-uplets varie d'un système à l'autre, cependant la structure générale décrite figure suivante revient fréquemment.



- Ainsi, un n-uplet commence par un en-tête précisant, entre autres, la longueur du n-uplet et éventuellement son type. Chaque attribut est précédé de sa longueur.

Dr BATOUMA NARKOY Master Informatique : BDA

7

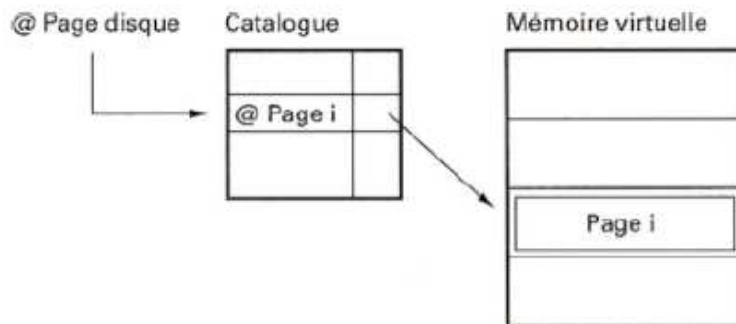
Mémoire virtuelle/mémoire tampon

- La plupart des SGBD gèrent leur propre zone de **mémoire tampon** encore appelée **mémoire virtuelle**, dans laquelle sont maintenues les données en cours de manipulation.
- Le principe d'un mécanisme de mémoire virtuelle consiste à découper l'espace mémoire en « cases » de la taille d'une page et à maintenir dans ces cases les pages contenant les données les plus fréquemment accédées, afin de favoriser les entrées-sorties.
- Une fonction importante du mécanisme de mémoire virtuelle consiste à décoder les adresses de pages en adresses mémoire.

Dr BATOUMA NARKOY Master Informatique : BDA

8

Mémoire virtuelle/mémoire tampon



Décodage d'une adresse de page en adresse mémoire

Dr BATOUMA NARKOY Master Informatique : BDA

9

Mémoire virtuelle/mémoire tampon

- La majorité des SGBD utilisent le même algorithme pour réquisitionner une case.
- Cet algorithme, connu sous le nom de LRU (*Least Recently Used*), consiste à réquisitionner une case libre s'il en existe une, ou la case la moins récemment utilisée.
 - Il est basé sur l'idée que la page la moins récemment utilisée a peu de chance d'être réutilisée à court terme. Il est à noter que, dans certains cas, l'algorithme LRU se comporte très mal et il faut appliquer l'algorithme inverse.
 - Certains SGBD utilisent en plus une technique de pré chargement, sachant que de nombreux algorithmes permettent de prévoir la liste des pages qui vont être accédées dans un futur proche

Dr BATOUMA NARKOY Master Informatique : BDA

10

Mémoire virtuelle/mémoire tampon

- Le rôle du mécanisme de mémoire virtuelle d'un SGBD est très similaire à celui des mécanismes de mémoire virtuelle offerts par les systèmes d'exploitation.
- La technique la plus efficace pour exploiter les mécanismes de mémoire virtuelle des systèmes d'exploitation consiste à calquer l'espace disque contenant la base de données dans l'espace mémoire virtuel géré par le système d'exploitation.
 - Cependant, cette solution est peu viable à l'heure actuelle, car les principaux systèmes d'exploitation offrent un espace de mémoire virtuelle limité en taille.
 - Cette contrainte de taille n'est pas acceptable pour de grosses bases de données. De plus, cet espace de mémoire virtuelle n'étant généralement pas prévu pour être persistant, il est difficile de garantir la fiabilité des données qu'il contient (dans une transaction base de données, les mises à jour doivent être effectuées en « tout ou rien » comme cela est expliqué par la suite ; une fois ces mises à jour validées, elles sont définitives).

Dr BATOUMA NARKOY Master Informatique : BDA

11

Mémoire virtuelle/mémoire tampon

- Enfin, le système d'exploitation ne peut pas tirer parti de la connaissance que l'on a des algorithmes du SGBD pour allouer au mieux les cases de la mémoire.
 - Par exemple, le système d'exploitation ne peut prévoir la liste des pages qu'il serait utile de pré charger pour optimiser les traitements du SGBD

Dr BATOUMA NARKOY Master Informatique : BDA

12

Mémoire virtuelle/mémoire tampon

- Le fait d'avoir deux mécanismes de mémoire virtuelle implantés l'un sur l'autre peut occasionner quelques dysfonctionnements (*le mécanisme de mémoire virtuelle d'un SGBD est généralement implanté par-dessus celui du système d'exploitation, c'est-à-dire que l'ensemble des cases gérées par le SGBD est contenu dans l'espace mémoire virtuel contenant le code et les données du SGBD*).
- En effet, l'accès à une page de données peut engendrer une double entrée-sortie dans les cas où le SGBD doit réquisitionner une case de l'espace de mémoire virtuelle du SGBD et où le système d'exploitation doit aussi réquisitionner une case de son propre espace de mémoire virtuelle pour charger la case du SGBD.

Dr BATOUMA NARKOY Master Informatique : BDA

13

Techniques de placement

Il y a au moins trois techniques de placement:

- Il s'agit des techniques de placement séquentiel et
- des techniques à accès direct basées sur les arbres-B
- ou des techniques à accès direct basées le hachage.
- Ces trois techniques sont comparées selon le nombre d'entrées-sorties nécessaire pour répondre à une requête faisant intervenir un critère simple.
- Deux types de critères simples sont envisagés :
 - ceux faisant intervenir une égalité du type « attribut = valeur », appelés *critère d'égalité* ;
 - ceux faisant intervenir une inégalité du type « attribut θ valeur » tel que θ soit un opérateur de comparaison parmi $<$, $<=$, $>$ et $>=$, appelés *critère d'inégalité*.

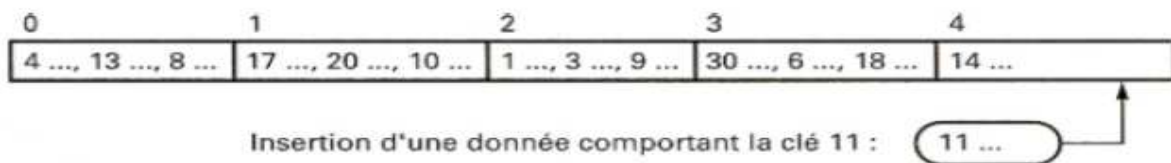
Dr BATOUMA NARKOY Master Informatique : BDA

14

Placement séquentiel

- Le *placement séquentiel*, consiste à placer chaque nouvelle donnée à la fin du segment, c'est-à-dire dans la dernière page constituant le segment.

Placement séquentiel:



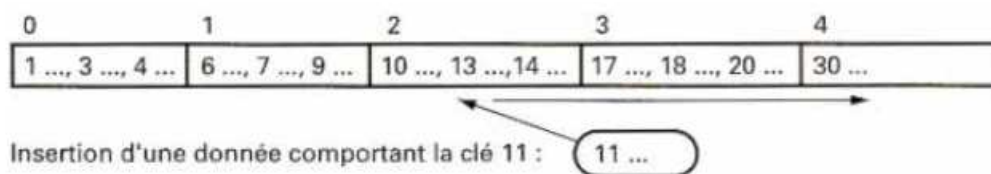
Dr BATOUMA NARKOY Master Informatique : BDA

15

Placement séquentiel

- Une amélioration de cette organisation consiste à **trier les données selon l'attribut le plus fréquemment accédé**. L'attribut sur lequel porte le tri est appelé *clé*. Cette organisation, appelée *placement séquentiel trié*

Placement séquentiel trié

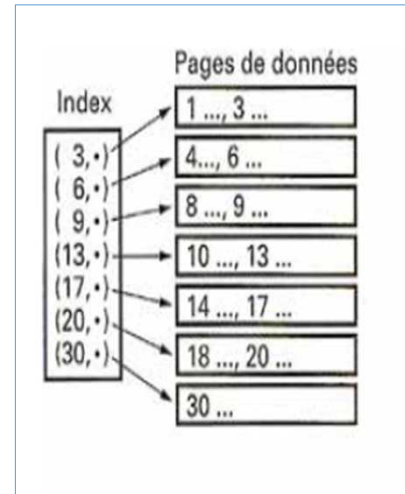


Dr BATOUMA NARKOY Master Informatique : BDA

16

Placement séquentiel indexé

- Le placement *séquentiel indexé* est une amélioration du placement séquentiel trié
- Cette méthode consiste à prévoir un peu de place libre dans chaque page de façon à faciliter les insertions, et à maintenir un index indiquant la plus grande clé stockée pour chaque page de façon à accélérer les recherches

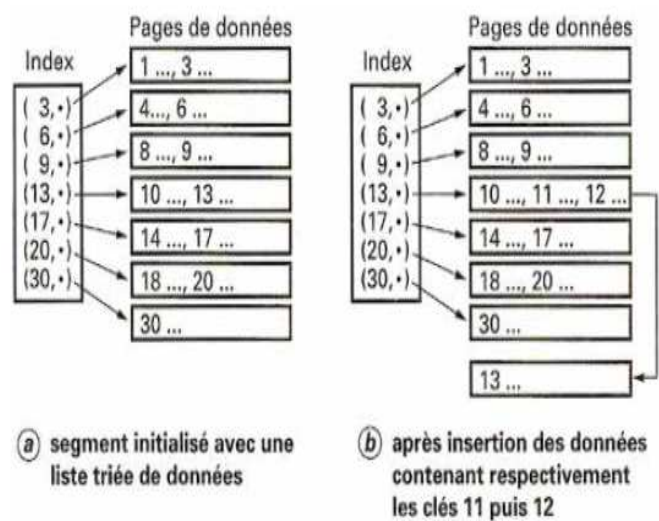


Dr BATOUMA NARKOY Master Informatique : BDA

17

Placement séquentiel indexé

- L'insertion d'une nouvelle donnée dans cette organisation se fait dans la page dont la plus grande clé est immédiatement supérieure à la valeur de la clé de la nouvelle donnée.
- Dans le cas où cette page est pleine, une page de débordement est chaînée à la page en question (**fig b**)



Dr BATOUMA NARKOY Master Informatique : BDA

18

Placement séquentiel indexé

- Le nombre d'entrées-sorties nécessaires pour répondre à un critère simple d'égalité dans cette organisation dépend du nombre de pages contenant l'index et du nombre moyen de pages de débordement pour une page donnée.
- Si l'index tient sur une seule page et s'il n'y a pas de pages de débordement, ce nombre est égal à deux, soit la lecture de la page d'index puis la lecture de la page susceptible de contenir la donnée recherchée.
- Dans le cas général, le nombre moyen d'entrées-sorties nécessaires pour répondre à un critère d'égalité peut être évalué à $(M/2) + (1 + D)/2$, M étant le nombre de pages constituant l'index et D correspondant au nombre moyen de pages de débordement.

Dr BATOUMA NARKOY Master Informatique : BDA

19

Placement séquentiel indexé

- Les données étant triées, le nombre d'entrées-sorties nécessaires pour répondre à un critère simple d'inégalité est égale à $N/2$ comme dans le cas du placement séquentiel trié.
- L'inconvénient principal de cette méthode est la gestion de pages de débordement qui rend aléatoire le nombre d'entrées-sorties nécessaires pour accéder à une donnée répondant à un critère simple d'égalité.
- Une autre difficulté, est la gestion d'un super index efficace, dans le cas où l'index ne tient pas dans une page.

Dr BATOUMA NARKOY Master Informatique : BDA

20

Structure d'index: arbre B

- Quand une table est volumineuse, un parcours séquentiel est une opération relativement lente et pénalisante pour l'exécution des requêtes, notamment dans le cas des jointures où ce parcours séquentiel doit parfois être effectué répétitivement.
- La création d'un *index* permet d'améliorer considérablement les temps de réponse en créant des chemins d'accès aux enregistrements beaucoup plus directs. Un index permet de satisfaire certaines requêtes (mais pas toutes) portant sur un ou plusieurs attributs (mais pas tous).

Structure d'index: arbre B

- L'index peut exister indépendamment de l'organisation du fichier de données, ce qui permet d'en créer plusieurs si on veut être en mesure d'optimiser plusieurs types de requêtes.
- En contrepartie la création sans discernement d'un nombre important d'index peut être pénalisante pour le SGBD qui doit gérer, pour chaque opération de mise à jour sur une table, la répercussion de cette mise à jour sur tous les index de la table.
- Un choix judicieux des index, ni trop ni trop peu, est donc un des facteurs conditionnant la performance d'un système.

B-arbre

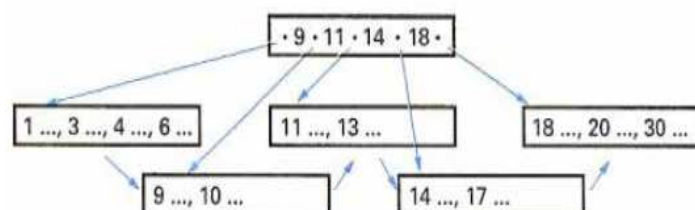
- Les **arbres-B** (arbres-balancés) constituent une amélioration de la technique de placement séquentiel indexé
- C'est une technique dynamique: dotée d'algorithmes d'insertion et de suppression
- Les **arbres-B+** peuvent être définis comme une technique de placement séquentiel trié, pour laquelle plusieurs niveaux d'index sont définis de façon à former une structure arborescente

Dr BATOUMA NARKOY Master Informatique : BDA

23

B-arbre

Pour simplifier la figure, nous considérons qu'une page constituant un nœud de l'arbre est pleine lorsqu'elle contient 4 clés. Ainsi un nœud contient au maximum 4 clés et au minimum 2 clés. Cet arbre contient un seul nœud racine. Les feuilles contiennent des données, matérialisées par les points de suspension. Les flèches en trait fin représentent le chaînage entre les feuilles.



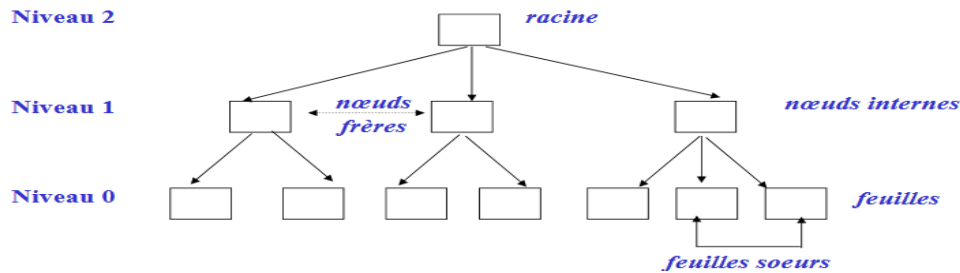
Dr BATOUMA NARKOY Master Informatique : BDA

24

B-arbre

Arbre B (R. Bayer et C. McCreight, 1972),
appelé également : *B-tree* ou arbre de *Bayer*

Le plus implanté : l'arbre B+



Dr BATOUMA NARKOY Master Informatique : BDA

25

B-arbre

Arbre B+ d'ordre m

- Tout nœud d'index a au **maximum m nœuds fils**
- un nœud possède au **minimum $\lceil m/2 \rceil$ fils**
- la racine possède au **minimum 2 fils**
- tout nœud d'index contient **k fils et (k-1) clés**

L'arbre est équilibré(balanced tree)

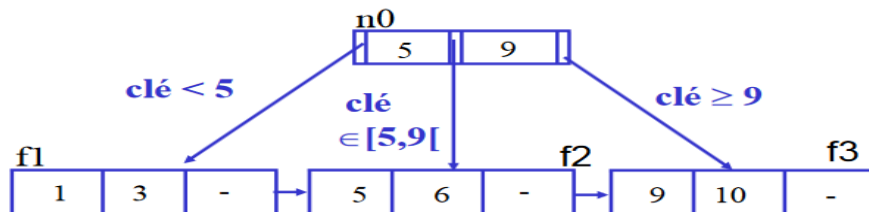
- tous les nœuds feuilles sont au même niveau
- la hiérarchie de l'arbre grossit par la racine :
tous les chemins de la racine aux nœuds feuilles ont la même longueur

Dr BATOUMA NARKOY Master Informatique : BDA

26

B-arbre

Arbre B+



Recherche de la clé 6 :

$6 \in [5,9[$ donc on va dans f2 et $6 \in f2$

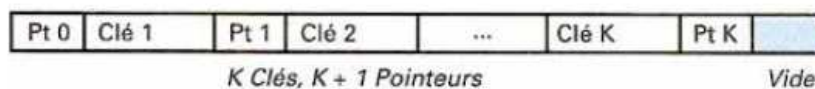
Dr BATOUMA NARKOY Master Informatique : BDA

27

B-arbre

Cette structure arborescente doit respecter les propriétés suivantes:

- Chaque page de cette structure est soit une feuille remplie de données triées, soit un nœud constituant une page d'index.
- Chaque nœud comporte **K clés** triées entourées de **K + 1 pointeurs**. Le i^e pointeur d'un nœud (Pti) référence la page dont les clés ont des valeurs comprises entre les valeurs de la $(i - 1)^e$ clé du nœud (Clé i - 1) et de la i^e clé du nœud (Clé i)



Dr BATOUMA NARKOY Master Informatique : BDA

28

B-arbre

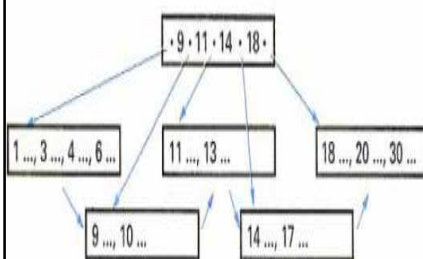
Cette structure arborescente doit respecter les propriétés suivantes (suite):

- Tous les chemins depuis la racine vers une feuille ont la même longueur que l'on appelle hauteur de l'arbre.
- Chaque page constituant l'arbre a un taux de remplissage compris entre 50 % et 100 %, sauf la racine dont le taux de remplissage peut aller de 0 % à 100 %.
- Les pages constituant les feuilles de l'arbre sont chaînées entre elles de façon à faciliter la lecture séquentielle des données.

Dr BATOUMA NARKOY Master Informatique : BDA

29

B-arbre



Insertion d'une donnée contenant la clé **8** dans l'arbre B+ précédent.

- L'algorithme d'insertion correspondant est récursif.
- Il consiste à **rechercher la feuille qui devrait contenir cette nouvelle donnée** (sur cet exemple, il s'agit de la feuille contenant déjà les clés 1, 3, 4 et 6).
- La page correspondante étant pleine, cette feuille est éclatée en deux et la clé médiane est remontée au niveau du nœud supérieur.
- Ce nœud étant lui-même plein, il est éclaté en deux ce qui provoque la création d'une nouvelle racine.
- Pour le détail de cet algorithme ainsi que pour l'algorithme de suppression nous renvoyons le lecteur aux nombreux livres publiés sur les bases de données.

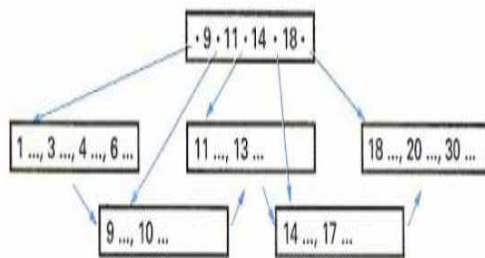
Dr BATOUMA NARKOY Master Informatique : BDA

30

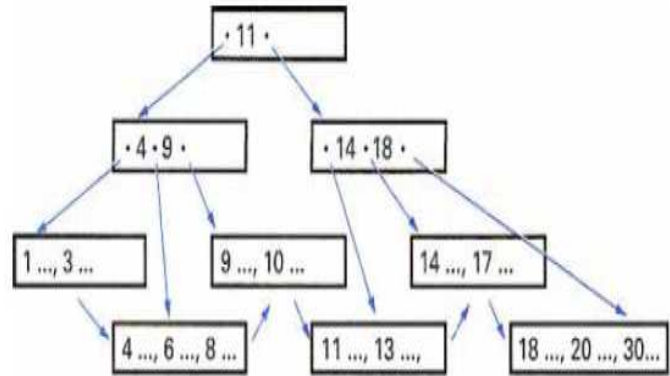
B-arbre

Insertion d'une donnée contenant la clé 8 dans l'arbre B+ précédent.

Avant insertion



Après insertion



Dr BATOUMA NARKOY Master Informatique : BDA

31

Hachage

Les techniques de hachage ont pour objectif de permettre l'accès direct à des données répondant à un critère simple d'égalité. Ces techniques sont basées sur les deux principes suivant :

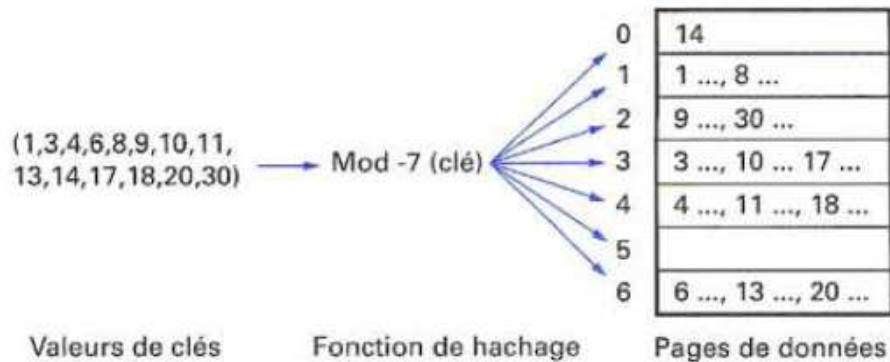
- l'espace destiné à recevoir les données à placer est découpé en P pages numérotées de 0 à P – 1 ;
- une fonction, appelée *fonction de hachage*, permet de convertir la valeur de clé de chaque donnée en un numéro compris entre 0 et P – 1, correspondant au numéro de la page dans laquelle la donnée est placée.

Dr BATOUMA NARKOY Master Informatique : BDA

32

Hachage

Placement de données dans 7 pages au moyen de la fonction modulo 7:

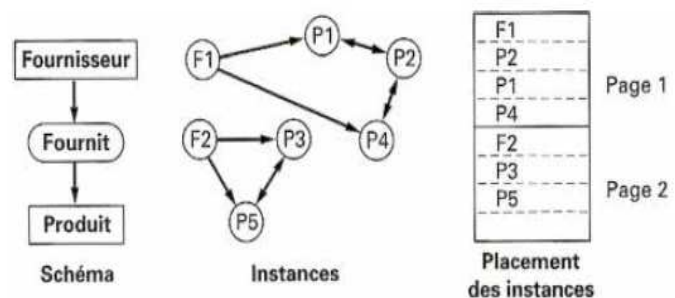


Dr BATOUMA NARKOY Master Informatique : BDA

33

Placement par proximité

- Le placement par proximité est une technique qui consiste à placer certaines données près des données avec lesquelles elles sont généralement utilisées.
- Cette technique a été introduite dans le cadre des SGBD réseaux pour regrouper dans les mêmes pages les données connectées entre elles.



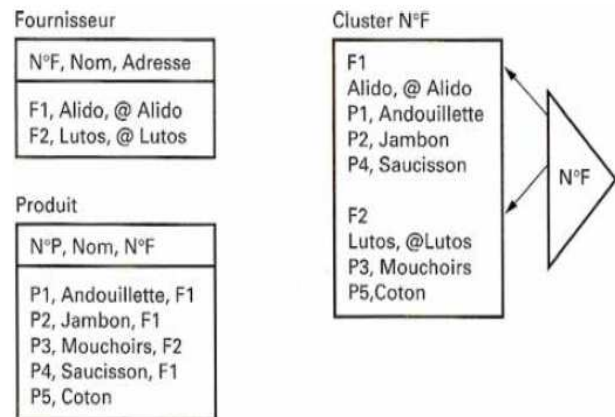
Placement par proximité dans un SGBD

Dr BATOUMA NARKOY Master Informatique : BDA

34

Placement par proximité

- Par exemple, le SGBD ORACLE offre la notion de **cluster** permettant de placer les n-uplets de plusieurs relations dans un même segment par rapport à un attribut commun.
- Ainsi, toutes les valeurs correspondant à l'attribut de placement du cluster sont stockées une seule fois, et les n-uplets partageant la même valeur d'attribut sont stockés bout à bout derrière cette valeur



Placement des tables Fournisseurs et Produit dans le cluster NoF