

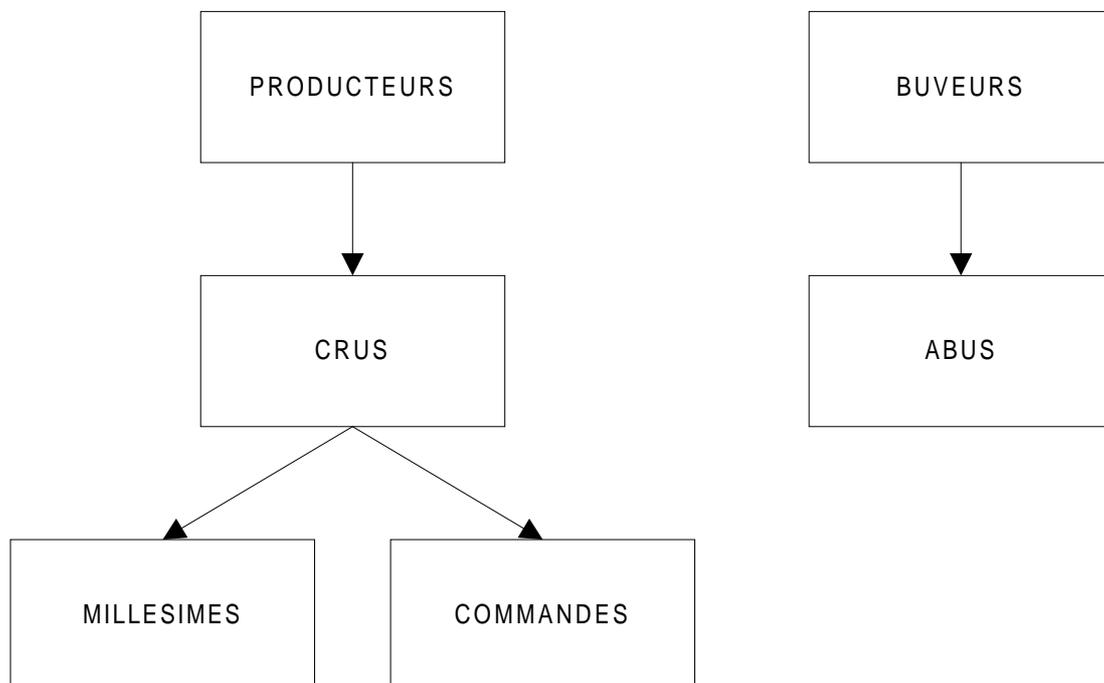
II. LES SYSTÈMES HIÉRARCHIQUES

Le modèle hiérarchique est assez bien adapté au monde réel, qui nous apparaît souvent au travers de hiérarchies. Beaucoup de systèmes sont encore basés sur ce modèle. Cependant, il n'existe pas de standard.

IMS d'IBM sur matériels IBM uniquement.

SYSTEM-2000 sur divers matériels : IBM, UNIVAC, CDC, CYBER.

1. Structure des bases de données hiérarchiques

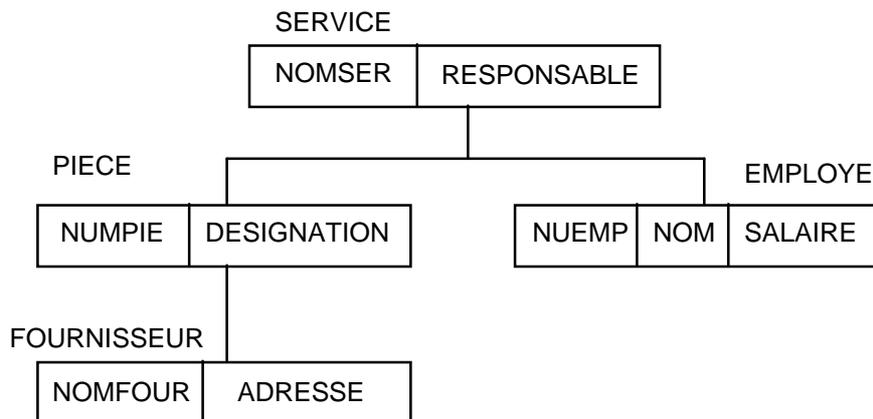


Exemple de base hiérarchique

a) Schéma hiérarchique

- Ensemble de *types d'enregistrements* R_1, R_2, \dots, R_n . Ex., le type CRUS, le type MILLESIME, etc.
- Ensemble de *liens* L_{ij} ($1 \leq i \leq n, 1 \leq j \leq n$). Ex. Un PRODUCTEUR produit des CRUS.
- 4 règles :
 - R1** : Il existe au plus un lien L_{ij} entre deux types d'enregistrements R_i et R_j .
 - R2** : Aucun lien L_{ii} n'est possible pour tout i .
 - R3** : Chaque lien L_{ij} correspond à une association 1-N (*parent – enfant*).
 - R4** : Le graphe structurel est un arbre ordonné appelé *schéma hiérarchique* (un *type racine* et des *types dépendants*).

b) Diagramme hiérarchique



Exemple de diagramme hiérarchique complet (champs inclus)

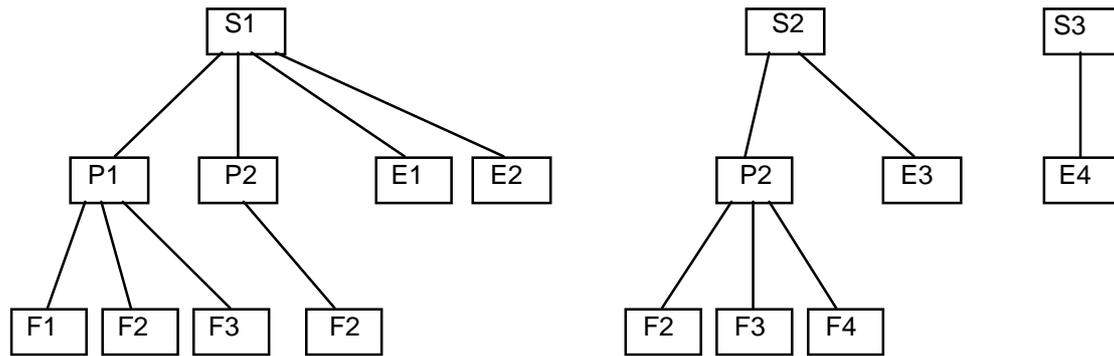
Définitions :

Schéma de base de données hiérarchique : un ou plusieurs schémas hiérarchiques

Schéma hiérarchique : un ensemble d'occurrences hiérarchiques

Occurrence hiérarchique : un arbre

Forêt des occurrences : juxtaposition des occurrences hiérarchiques



Forêt d'occurrences hiérarchiques pour le diagramme hiérarchique précédent

c) Propriétés d'un schéma hiérarchique

- Le type racine ne doit pas participer comme enfant par ailleurs.
- Chaque type, sauf la racine, figure comme enfant dans une et une seule liaison (chaque type, sauf la racine possède un et un seul type parent).
- Un type peut jouer le rôle de parent pour plusieurs autres types (les enfants ordonnés de gauche à droite).
- Un type qui ne figure pas comme parent est appelé feuille du schéma hiérarchique.

d) Forme linéaire d'un schéma hiérarchique

Certains langages de manipulation reposent sur une exploration linéaire.

Séquence hiérarchique : parcours en *préordre* du schéma hiérarchique.

```

PROCEDURE Préordre(X) / X est la racine de l'arbre /
Début
  Output(X)
  Pour chaque enfant Y de X dans l'ordre gauche-droite faire
    Préordre(Y)
Fin
  
```

Ex. Séquence hiérarchique SERVICE, PIECE, EMPLOYE, FOURNISSEUR.

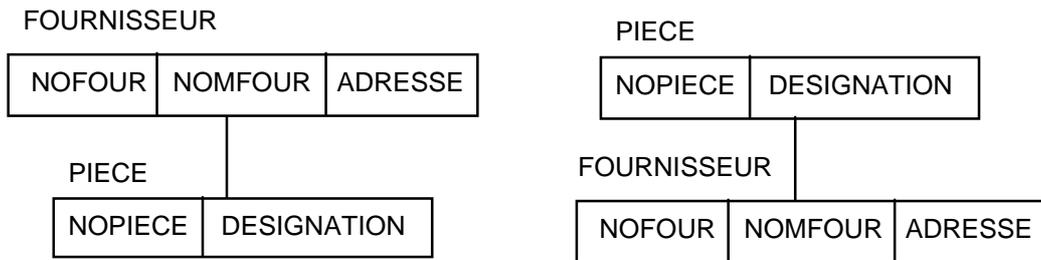
Chemin hiérarchique : séquence de nœuds N_1, N_2, \dots, N_i où N_1 est la racine et où N_j est un enfant de N_{j-1} pour $j = 2, 3, \dots, i$.

Ex. Chemin hiérarchique SERVICE, PIECE, FOURNISSEUR.

e) *Représentation des associations M-N à l'aide d'un schéma hiérarchique*

Exemple : Association FOURNISSEURS-PIECES indiquant les pièces approvisionnées par un fournisseur et les fournisseurs qui approvisionnent une pièce.

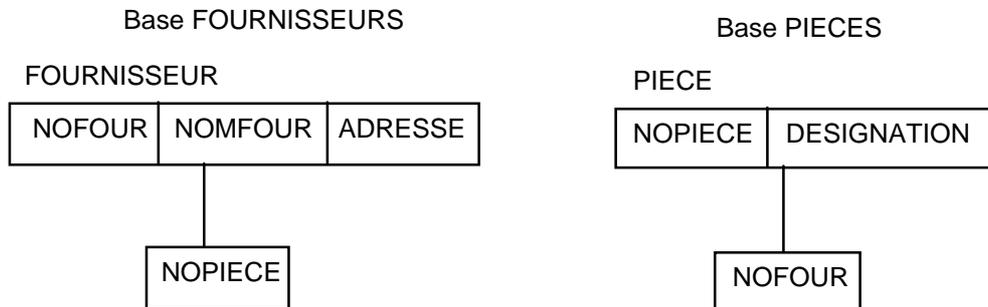
Représentations non symétriques



Deux représentations non symétriques d'une association M-N

Les performances d'accès ne sont pas les mêmes pour les deux types.

Représentation symétrique

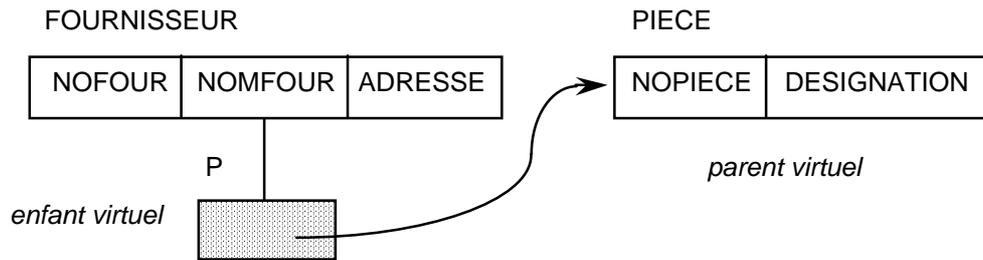


Représentation symétrique d'une association M-N

- ⇒ Redondance des identifiants
- ⇒ Manipulation conjointe par le programmeur

Liaison virtuelle parent-enfant

Utilisation d'un pointeur au sein d'un même schéma hiérarchique ou entre des schémas hiérarchiques différents.



Représentation d'une association M-N par liaison virtuelle

Symétrisation possible en utilisant un segment dépendant du type PIECE et en installant un deuxième pointeur vers le type FOURNISSEUR.

P peut contenir des *données d'intersection*.

2. Définition de bases hiérarchiques

a) Contraintes du modèle hiérarchique

- Une occurrence qui n'est pas une occurrence de la racine, ne peut pas exister dans la base sans être reliée à une occurrence parent réel.
- Toute occurrence autre que celle de la racine ne peut être insérée que si son parent réel est déjà présent dans la base.
- La suppression d'un parent réel entraîne automatiquement la suppression de tous ses enfants réels et de ses descendants réels.
- Une occurrence ne peut être accédée qu'à travers un chemin hiérarchique.

b) Définition d'une base hiérarchique

Cf. exemple ci-contre.

- Ligne 1 : nomme la base de données et précise l'organisation physique.
- Lignes 2 à 4 : description du segment racine SERVICE. Option SEQ : tri des occurrences de ce segment selon l'ordre ascendant des valeurs de NOMSER. Valeurs identiques interdites.
- Lignes 5 à 7 : description du segment PIECE.

```

1  DBD NAME=SERVICEDB, ACCESS=HDAM
2  SEGM NAME=SERVICE, BYTES=30
3  FIELD NAME=(NOMSER, SEQ), BYTES=15, START=1
4  FIELD NAME=RESPONSABLE, BYTES=15, START=16
5  SEGM NAME=PIECE, BYTES=40, PARENT=SERVICE
6  FIELD NAME=(NUMPIE, SEQ), BYTES=10, START=1
7  FIELD NAME=DESIGNATION, BYTES=30, START=11
8  SEGM NAME=FOURNISSEUR, BYTES=40, PARENT=PIECE
9  FIELD NAME=(NOMFOUR, SEQ, M), BYTES=15, START=1
10 FIELD NAME=ADRESSE, BYTES=25, START=16
12 SEGM NAME=EMPLOYE, BYTES=45, PARENT=SERVICE
13 FIELD NAME=(NUEMP, SEQ), BYTES=10, START=1
14 FIELD NAME=NOM, BYTES=25, START=11
15 FIELD NAME=SALAIRE, BYTES=10, START=36
16 DBDGEN
17 FINISH
18 END

```

Exemple de description avec IMS/DL1

- Lignes 8 à 10 : description du segment FOURNISSEUR. Option M + SEQ : clé non unique.
- Lignes 12 à 15 : description du segment EMPLOYE.
- Lignes 16 à 18 : indication de fin de description et fin de programme.

3. Manipulation de bases hiérarchiques

Quelques exemples de transactions (syntaxe de IMS/DL1).

a) Accès direct

Caractéristiques de la pièce 118 :

```
GET UNIQUE SERVICE PIECE (NUMPIE=118)
```

Il faut indiquer le chemin hiérarchique complet. S'il existe plusieurs pièces portant le même numéro, c'est la première rencontrée qui est sélectionnée.

b) Accès séquentiel

Caractéristiques de tous les fournisseurs à partir de la pièce 118 :

```
GET UNIQUE SERVICE PIECE (NUMPIE=118) FOURNISSEUR
  while DB-STATUS=0 do
    begin
      GET NEXT FOURNISSEUR
    end
```

c) Accès séquentiel sous un même parent

Noms et adresses des fournisseurs de la pièce 118 pour le service SECURITE :

```
GET UNIQUE SERVICE (NOMSER=SECURITE) PIECE (NUMPIE=118)
  while DB-STATUS=0 do
    begin
      GET NEXT WITHIN PARENT FOURNISSEUR
    end
```

d) Insertion

Insérer un nouveau fournisseur pour la pièce 118 :

```
INSERT SERVICE PIECE (NUMPIE=118) FOURNISSEUR
```

e) Suppression

Supprimer la pièce 118 et ses descendants :

```
GET HOLD UNIQUE SERVICE PIECE(NUMPIE=118)
DELETE
```

f) Remplacement

Modifier la désignation de la pièce 118 :

```
GET HOLD UNIQUE SERVICE PIECE (NUMPIE=118)
  Effectuer la modification dans la zone E/S
REPLACE
```