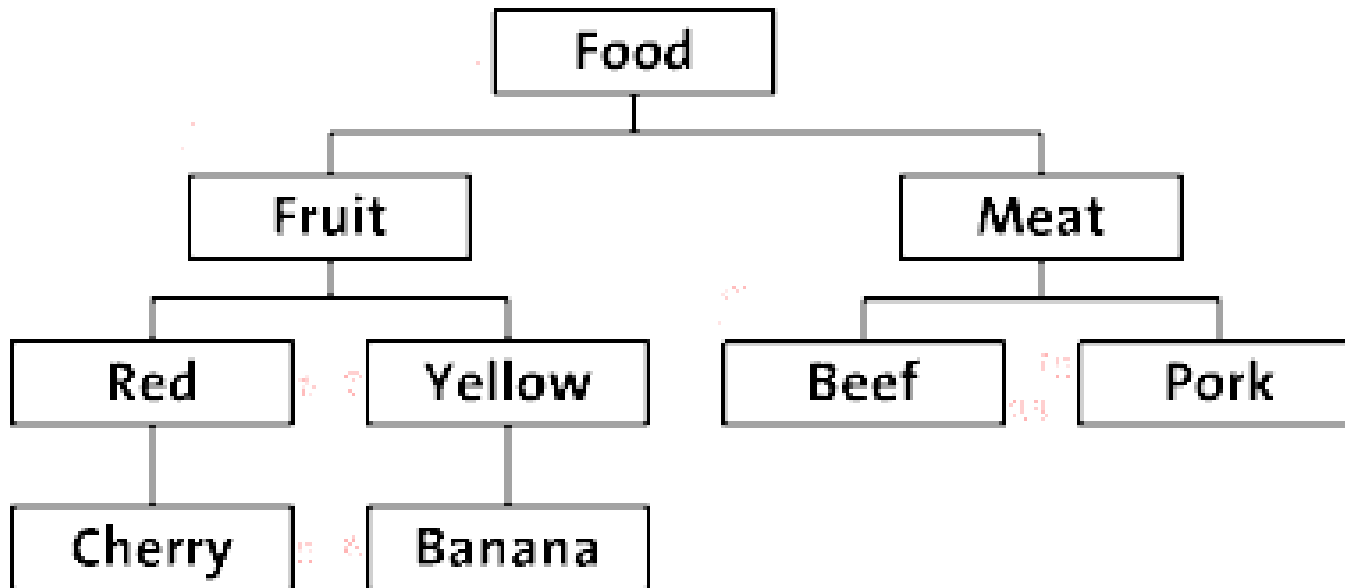


Structures Arborescente

Introductions aux Bases de Données
Nathanaël Martel

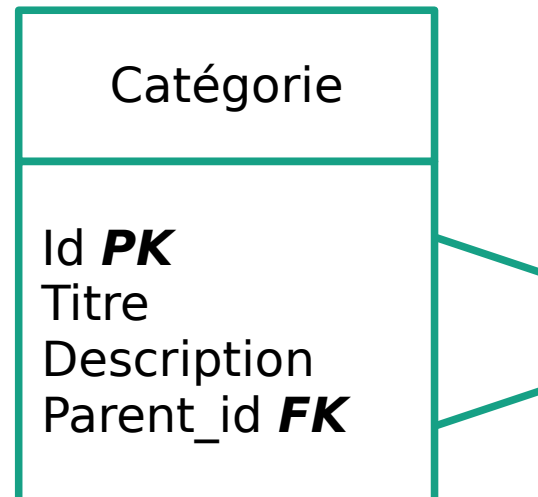
Arbre

- **Problématique : comment mettre sous forme de table une structure d'arbre ?**



Arbre simple

- Un enregistrement par branche.
- Ajout d'une liaison sur la table elle même, généralement avec une clé étrangère (!?) nommé **parent_id** liée à la clé primaire de la même table.



Arbre simple

- Récupération du parent :

```
SELECT * FROM `categorie`  
WHERE `id` IN (SELECT `parent_id` FROM `categorie` WHERE  
id=287) ;
```

```
SELECT * FROM `categorie` WHERE id=12 ;
```

- Récupération des enfants :

```
SELECT * FROM `categorie`  
WHERE `parent_id` IN (SELECT `id` FROM `categorie` WHERE  
id=287) ;
```

```
SELECT * FROM `categorie` WHERE parent_id=12 ;
```

Arbre simple

- **Facile à mettre en place et à maintenir**
- **Problème :**
 - Quelle est la «profondeur» de la branche ?
 - Comment récupérer d'un coup (une seule requête) tous les parents ?
 - Comment savoir s'il y a des enfants sans faire de requête ?
 - Comment récupérer d'un coup (une seule requête) tous les enfants ?
 - Comment ordonner les enfants les uns avec les autres ?

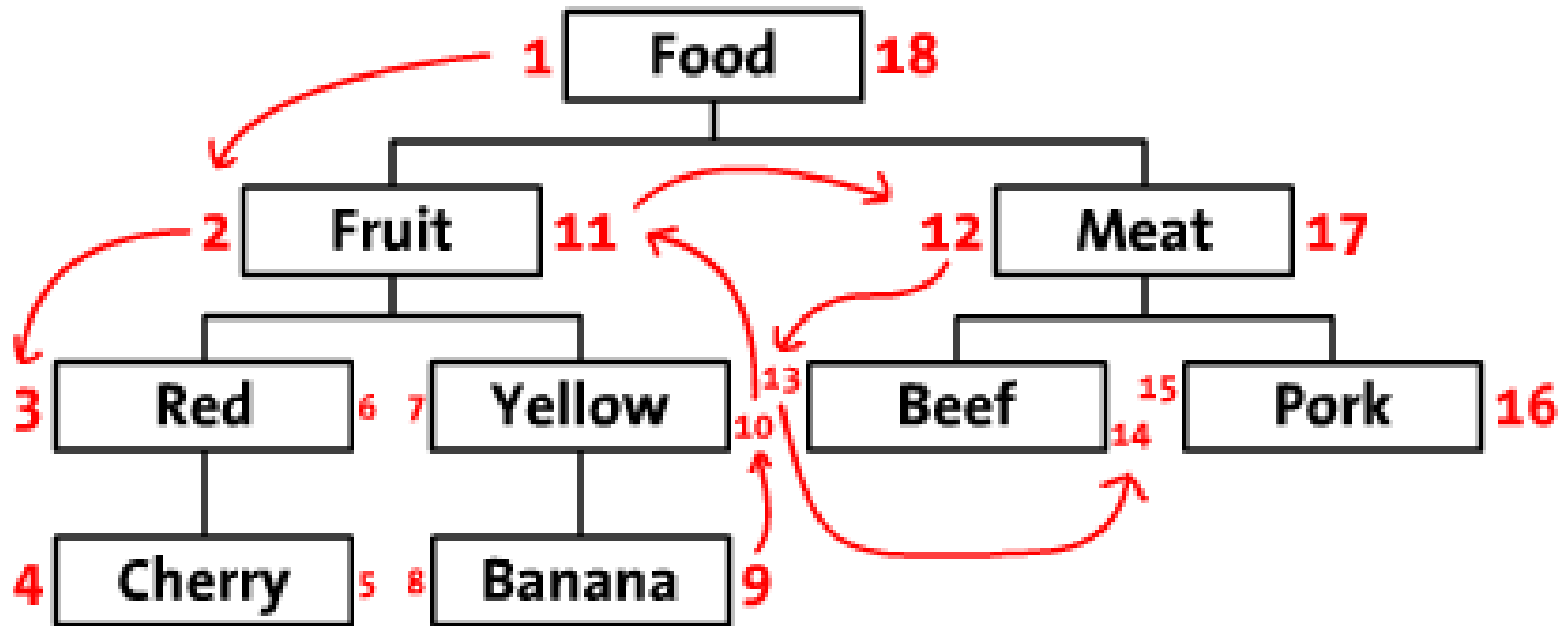
Arbre avec profondeur

- Gestion de la profondeur : ajout d'un champs indiquant la profondeur, par exemple **depth**.
- C'est un entier, donnant le nombre de parent.
- Problème : il peut y avoir une incohérence entre les deux champs.

Arbre complexe

- Ajout de deux champs pour chaque nœud indiquant son extrême gauche et son extrême droite.
- Par exemple : **nleft** et **nright**

Arbre complexe



Arbre complexe

Problèmes :

- Il peut y avoir une incohérence avec les autres champs.
- Le champs **parent_id** est-il nécessaire ?

Arbre complexe

- Récupérer tous les parents d'un nœud donné

```
SELECT * FROM `ps_category`  
WHERE nleft <= 559 AND nright >= 568  
ORDER BY nleft;
```

- Récupérer tous les enfants en respectant l'ordre

```
SELECT REPEAT('***', level_depth), id_category,  
level_depth, nleft, nright  
FROM `ps_category`  
WHERE nleft >= 559 AND nright <= 568  
ORDER BY nleft;
```

Conclusion

Pour des arbres simple, un champs parent_id suffit.
Pour des arbres plus complexe, il faut utiliser une structure plus complexe mais plus adapté.