

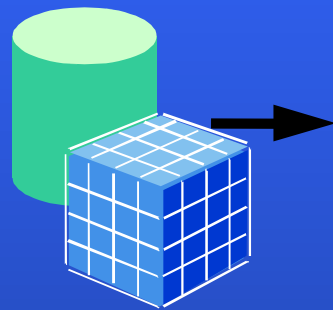
Oracle Décisionnel : Modèle OLAP et Vue matérialisée

SOMMAIRE

- Introduction
- Le modèle en étoiles
- Requêtes OLAP
- Vue matérialisée
- Fonctions Roll up et Cube
- Application

Introduction

Data Warehouse



Données

Stockage des
données détaillées (et
agrégées)

Moteur OLAP



Traitements

Aide à la décision

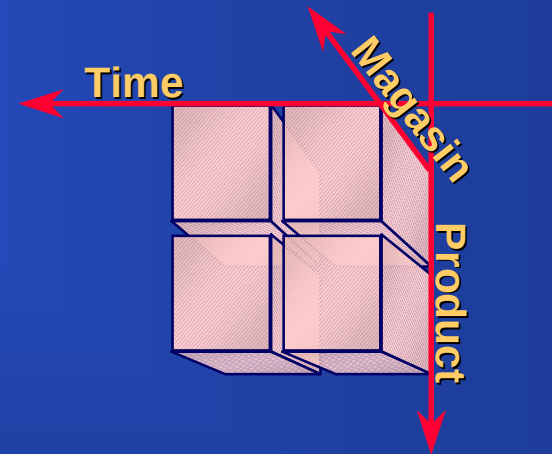


Présentation

Rapports
Multi-Dimensionnel

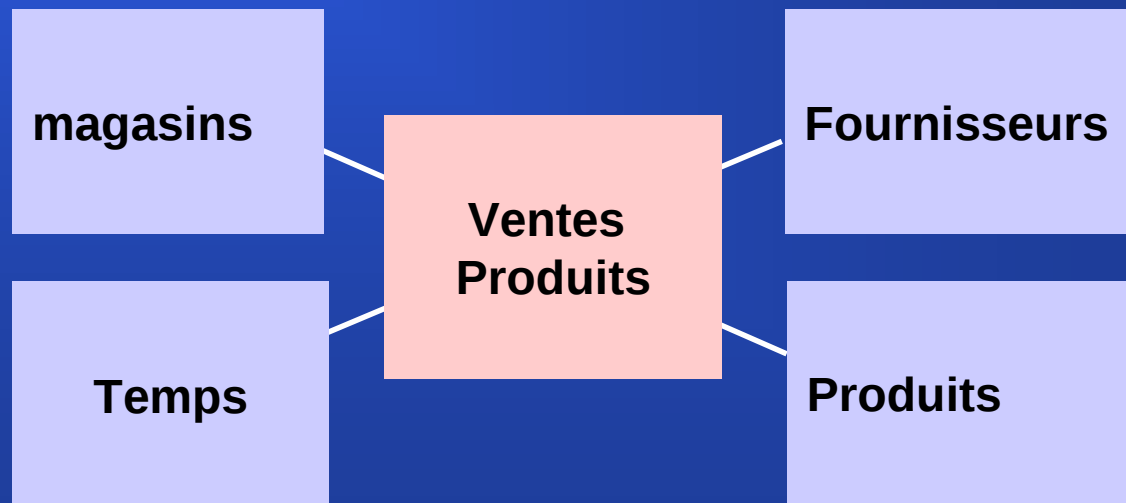
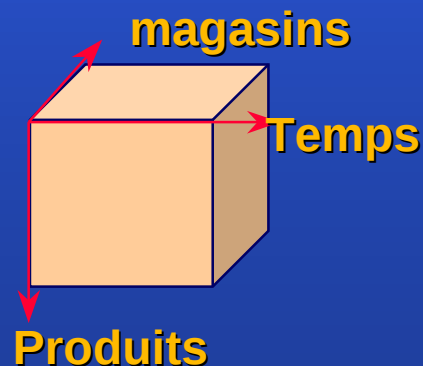
Modèle en étoiles

- Le modèle multidimensionnel ou modèle en étoiles (ou OLAP) est une méthodologie de structuration des données dédiée au reporting et à l'analyse.
- Les données sont stockées sous forme de cube.
- La donnée se trouve à l'intersection de n dimensions

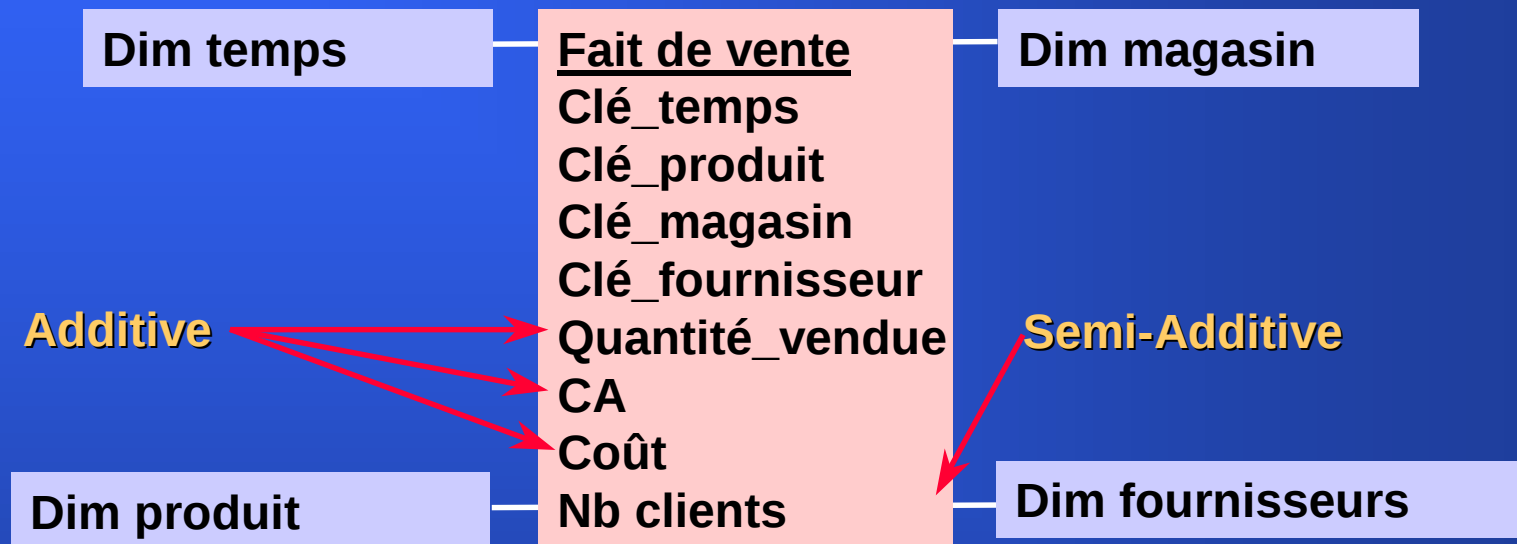


Caractéristiques du modèle en étoiles

- Un cube est caractérisé par un fait qui se produit
- Les faits sont quantifiés par des indicateurs
- Les dimensions sont les axes d'analyse.
- Chaque dimension est associée à une hiérarchie

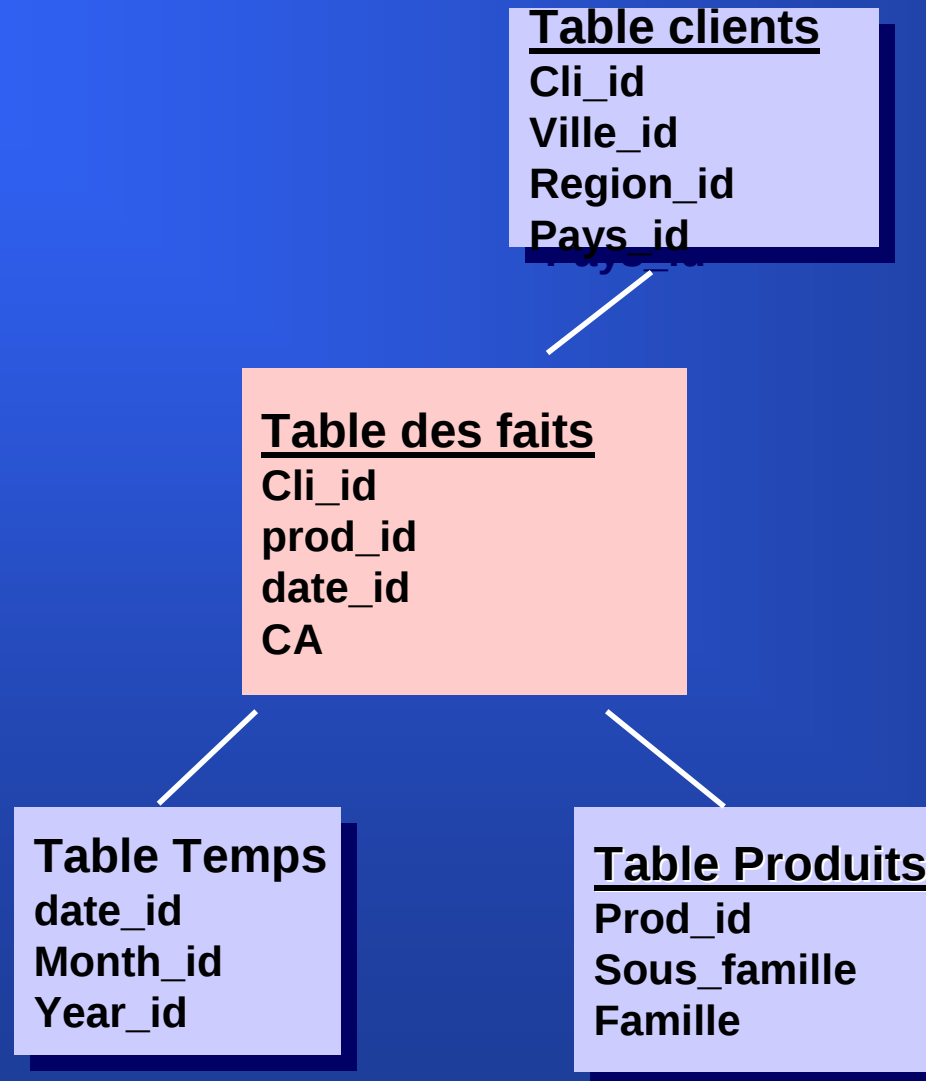


La table de faits



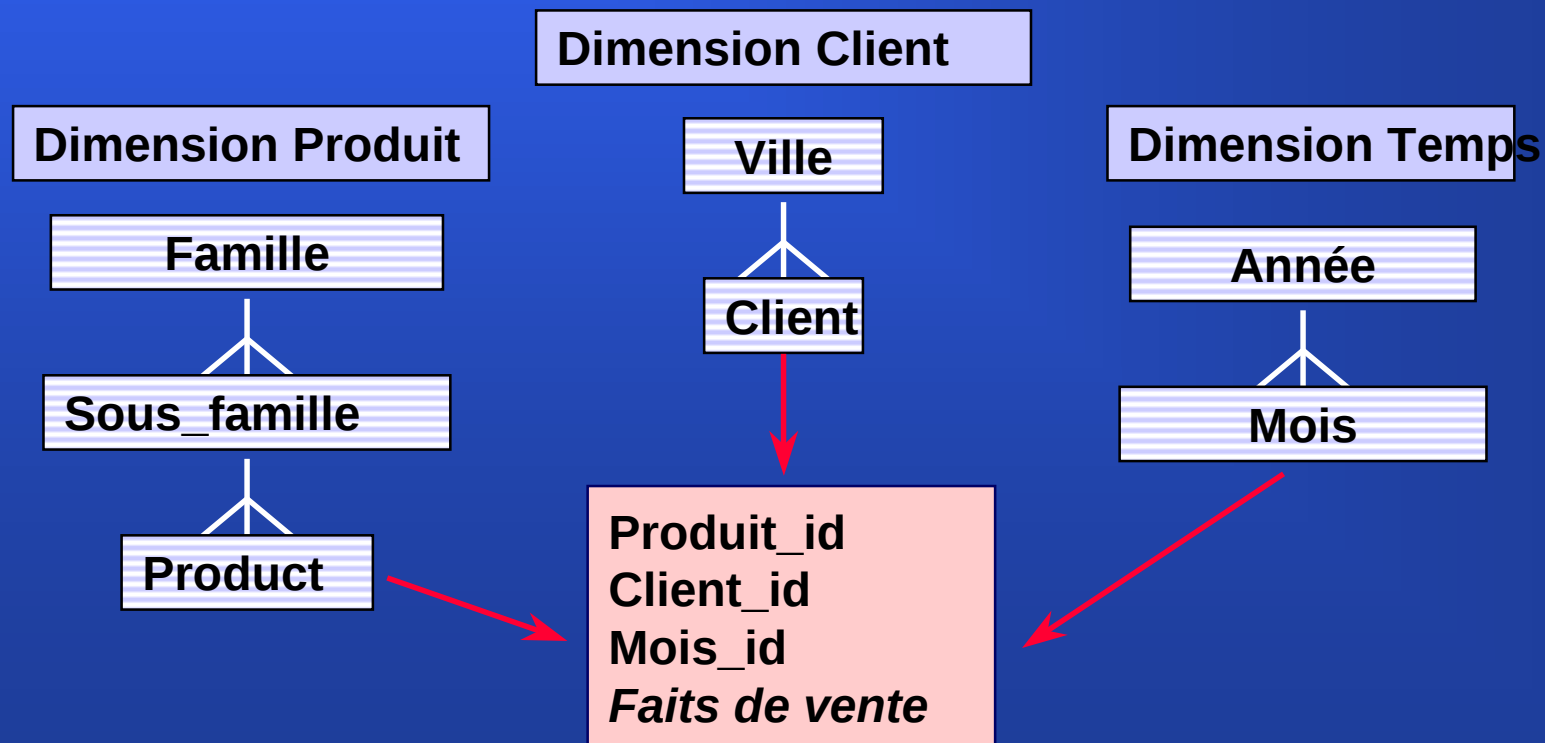
Coût par Produit + Magasin + Temps
Coût par Produit + Fournisseur + Temps
CA par Produit + Magasin
CA par Produit + Temps . . .

Les tables de dimension



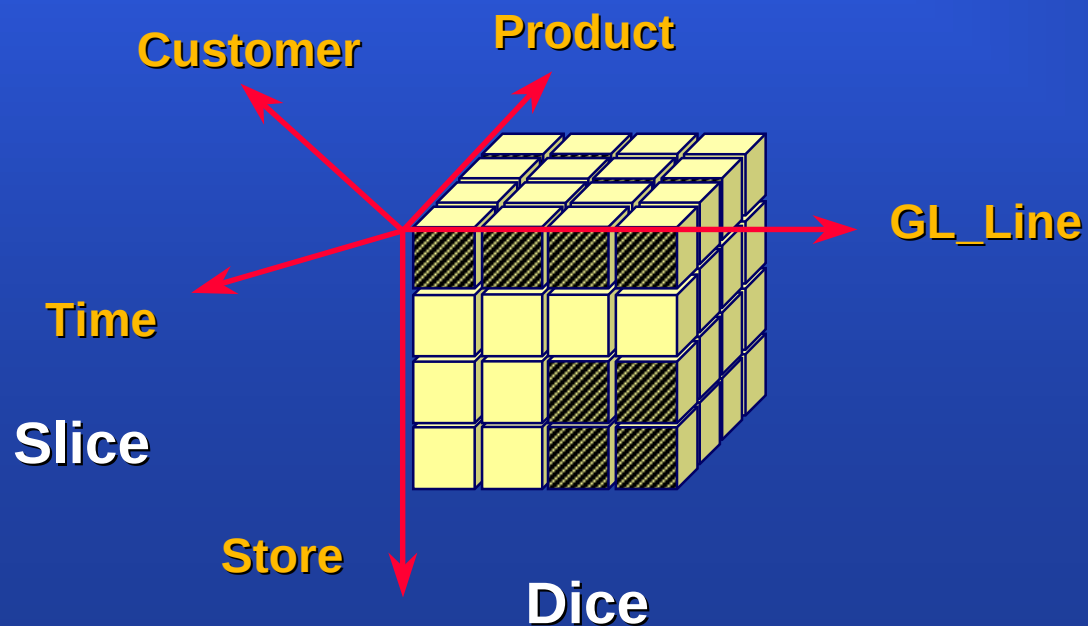
Les hiérarchies

- Chaque dimension est associée à une hiérarchie
- La relation est de type (1;n) entre le père et le fils
- Chaque niveau de dimension correspond à une colonne

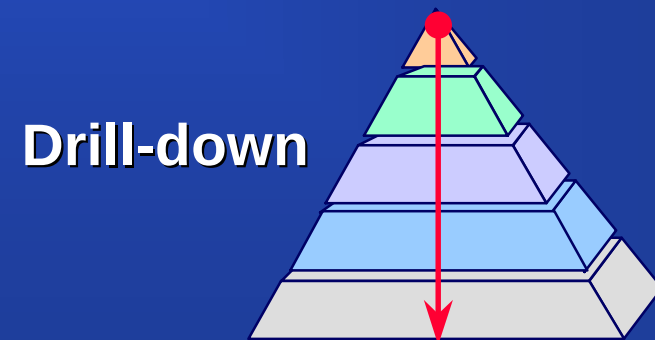
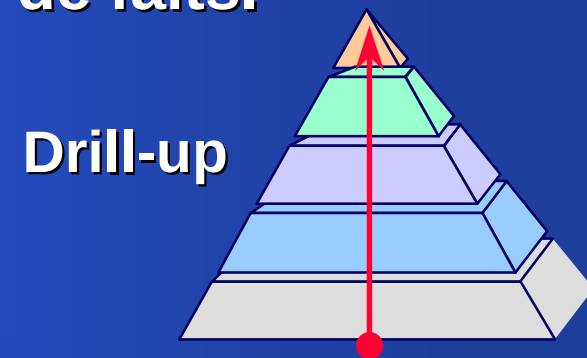


Requête OLAP

Une requête OLAP consiste à extraire des données du cube en appliquant des critères sur les tables de dimension et en choisissant un ou plusieurs indicateurs contenus dans la table de faits.



D BILEK



Caractéristiques requêtes OLAP

Les requêtes OLAP sont complexes et nécessitent des calculs d'agrégats et de multiples jointures

Exemple : on souhaite connaître le CA par magasin, par mois et par produit pour l'année 2001 sur la région parisienne.

Afin d'augmenter la performance de ces requêtes on peut stocker ou matérialiser tout ou une partie du cube.

Cela veut dire que chaque niveau d'agrégation est pré-calculé en combinaison avec les autres dimensions

Vues matérialisée

Une vue matérialisée est un objet de la BDD qui sert à stocker des données résumées et des résultats précalculés dans une vue.

L'optimiseur de requêtes s'appuie alors sur ces vues dès qu'il le peut afin d'éviter des opérations de jointure et d'agrégation coûteuses.

Caractéristiques Vues matérialisées

- stockée dans la même BDD que les tables de faits et de dimension
- transparente pour l'utilisateur et l'application
- mise à jour lorsque les tables sous-jacentes sont modifiées
- 3 types de vues matérialisées
 - Sans agrégats
 - Avec agrégats
 - Imbriquées

Syntaxe

Exemple Analyse des ventes par produit

```
CREATE MATERIALIZED VIEW product_sales_mv  
BUILD Deferred / Immediate  
REFRESH Complete / Fast / Force  
ON DEMAND / ON COMMIT  
ENABLE QUERY REWRITE  
AS SELECT p.prod_name, SUM(CA) AS CA  
FROM sales s, products p  
WHERE s.prod_id = p.prod_id  
GROUP BY p.prod_name;
```

Fonction Roll up

Roll up permet de calculer des sous totaux en allant du plus détaillé vers le plus général. la fonction crée des sous-totaux sur n+1 niveaux pour n colonnes de groupage. Pour chaque niveau un total est calculé puis un total général

```
SELECT annee, mois, prod, sum( CA )  
FROM sales, time  
WHERE sales.date_achat=time.date  
And Mois='Janv' and year='2001'  
GROUP BY  
ROLLUP(annee,mois,prod)
```

Année	Mois	Prod	CA
2001	Jan	A	200
2001	Jan	B	400
2001	Jan		600
2001			600
			600

Fonction Cube

Cube est une fonction qui permet de calculer toutes les combinaisons de sous totaux pour un groupe de dimensions

```
SELECT annee, mois, prod sum( CA )  
FROM sales, time WHERE  
sales.date_achat=time.date  
And Mois='Janv' and year='2001'  
GROUP BY  
CUBE(annee,mois,prod)
```

Année	Mois	Prod	CA
2001	Jan	A	200
2001	Jan	B	400
2001	Jan		600
2001		A	200
2001		B	400
	Jan	A	200
	Jan	B	400
		A	200
		B	400
	Jan		600
2001			600
			600

Exemple

```
CREATE MATERIALIZED VIEW cube
```

```
( ... )
```

```
AS SELECT
```

```
Pays, région, ville, produit, sous_famille, famille, année, mois, date
```

```
Sum(CA), sum(cout)
```

```
FROM sales s, produits p, temps t, clients c
```

```
WHERE s.idprod = p.idprod and s.idcli=c.idcli and s.idtps=t.idtps
```

```
GROUP BY
```

```
Rollup ( famille, sous_famille, produit)
```

```
Rollup ( pays, region, ville )
```

```
Rollup ( année, mois, date )
```

Bibliographie

- Oracle 8i Notions fondamentales (Campus Press)
- Oracle 9i OLAP User's Guide
- Oracle 9i Data Warehousing guide
(www.cs.umb.edu/cs634/ora9idocs)