# Les SGBDs Décisionnels

Didier DONSEZ

Université Joseph Fourier (Grenoble 1)
IMA – IMAG/LSR/ADELE

**Didier.Donsez@imag.fr**

# Plan

- Limites du SQL en décisionnel
- Nouvelles fonctions et types de données
- Aspect Systèmes
  - Index Bitmap
  - RAID
- Benchmarks

# Limites du SQL et des SGBDs « Transactionnels »

orienté vers le OLTP

- Interface
  - Manque d ’expressivité des requêtes SQL pour l’OLAP
- Système
  - Structures et Algorithmes inadaptées à la charge de travail de l ’OLAP

# Systèmes

- MOLAP
  - SGBD Spécialisé
- ROLAP
  - SGBD Relationnel
- H-OLAP
  - SGBD Relationnel avec des opérateurs et algorithmes adéquats
    - StarJoin, Index Bitmap, ...
    - GROUP BY CUBE, …

# Nouvelles Fonctions

- BREAK BY (SAS)
- RANK
  - Rang d'une ligne par rapport à un agrégat
- TOP / BOTTOM
  - Requête de type « Top Ten »
    - Les 10 meilleurs / Les 10 moins bons
- Echantillonnage (Informix Online XPS)
  - Requête effectuée sur un échantillon de données tiré aléatoirement (preview)
    - Limite de temps de calcul pour une approximation du résultat
- Extension du Group By (SQL99)
  - Grouping Sets, Rollup, Cube
- Fenêtre mouvante pour les agrégat : Windows
  - Exemple : moyenne et déviance sur le jour courant et les 3 jours précédents

# GROUP BY GROUPING SETS

- Partitionnement selon plusieurs dimensions
- Exemple

**SELECT #Client, #Produit, SUM(Montant) FROM Ventes**

**GROUP BY GROUPING SETS ((#Client), (#Produit))**

- est équivalent à

**(SELECT #Client, NULL, SUM(Montant) FROM Ventes GROUP BY #Client)**

**UNION**

**(SELECT NULL, #Produit, SUM(Montant) FROM Ventes GROUP BY #Produit)**

# GROUP BY ROLLUP

- Réduire progressivement

- Exemple

**SELECT #Client, #Produit, SUM(Montant) FROM Ventes**
**GROUP BY ROLLUP (#Client, #Produit)**

- est équivalent à

**SELECT #Client, #Produit, SUM(Montant) FROM Ventes**
**GROUP BY GROUPING SETS ((#Client,#Produit), (#Client), ())**

- est équivalent à

**(SELECT #Client, #Produit , SUM(Montant) FROM Ventes GROUP BY #Client, #Produit)**
**UNION**
**(SELECT #Client, NULL, SUM(Montant) FROM Ventes GROUP BY #Client)**
**UNION**
**(SELECT NULL, NULL, SUM(Montant) FROM Ventes)**

# GROUP BY ROLLUP

*SELECT MONTH(SALES_DATE), REGION, SALES_MGR, SUM(SALES)*

*FROM SALES WHERE YEAR(SALES_DATE) = 1996*

*GROUP BY* ***ROLLUP (MONTH(SALES_DATE), REGION, SALES_MGR)***

| MONTH | REGION | SALES_MANAGER | SUM(SALES) |
|---|---|---|---|
| *April* | *Central* | *Chow* | *25000* |
| *April* | *Central* | *Smith* | *15000* |
| *April* | *Central* | *-* | *40000* |
| *April* | *North* | *-* | *15000* |
| *April* | *North* | *-* | *15000* |
| *April* | *-* | *-* | *55000* |
| *May* | *Central* | *Chow* | *25000* |
| *May* | *Central* | *-* | *25000* |
| *May* | *North* | *Smith* | *15000* |
| *May* | *North* | *-* | *15000* |
| *May* | *-* | *-* | *40000* |
| *-* | *-* | *-* | *95000* |

# GROUP BY CUBE

- Partitionnement selon tous les sous-ensembles possibles de Grouping Sets

```
SELECT #Client, #Produit, #Fournisseur, SUM(Montant) FROM Ventes
GROUP BY CUBE (#Client, #Produit, #Fournisseur)
```

- est équivalent à

```
SELECT #Client, #Produit, SUM(Montant) FROM Ventes
GROUP BY GROUPING SETS (
()                                  -- total des ventes
(#Client),                          -- total des ventes par Client
(#Fournisseur),                     -- total des ventes par Forunisseur
(#Produit),                         -- total des ventes par Produit
(#Client, #Fournisseur)             -- total des ventes par Client et par Fournisseur
(#Client, #Produit),                -- total des ventes par Client et par Produit
(#Produit, #Fournisseur), -- total des ventes par Produit et par Fournisseur
(#Client, #Produit, #Fournisseur) -- total des ventes par Client, ¨Produit et Fournisseur
)
```

# Fenêtre glissante

- But: cumul, moyenne, dérivation sur une fenêtre glissante du temps

**SELECT Sf.Region, Sf.Month, Sf.Sales, AVG (Sf.Sales) OVER (**

**PARTITION BY Sf.Region ORDER BY Sf.Month ASC ROWS 1 PRECEDING**

**) AS Moving_avg FROM SalesFact AS Sf ORDER BY Sf.Month ASC;**

| Region | Month | Sales | Moving_Avg |
|---|---|---|---|
| Eastern | Oct, 1994 | 27497 | 27497 |
| Eastern | Nov, 1994 | 24168 | 25832 |
| Eastern | Dec, 1994 | 27801 | 25984 |
| Eastern | Jan, 1995 | 25991 | 26896 |
| Eastern | Feb, 1995 | 25968 | 25979 |
| Eastern | Mar, 1995 | 23610 | 24789 |
| Mid Atlantic | Oct, 1994 | 7150 | 7150 |
| Mid Atlantic | Nov, 1994 | 7586 | 7368 |
| Mid Atlantic | Dec, 1994 | 6164 | 6875 |
| Mid Atlantic | Jan, 1995 | 6051 | 6108 |
| Mid Atlantic | Feb, 1995 | 4299 | 5175 |
| Mid Atlantic | Mar, 1995 | 6283 | 5291 |

# MS MDX *(Multidimensional Expression)*

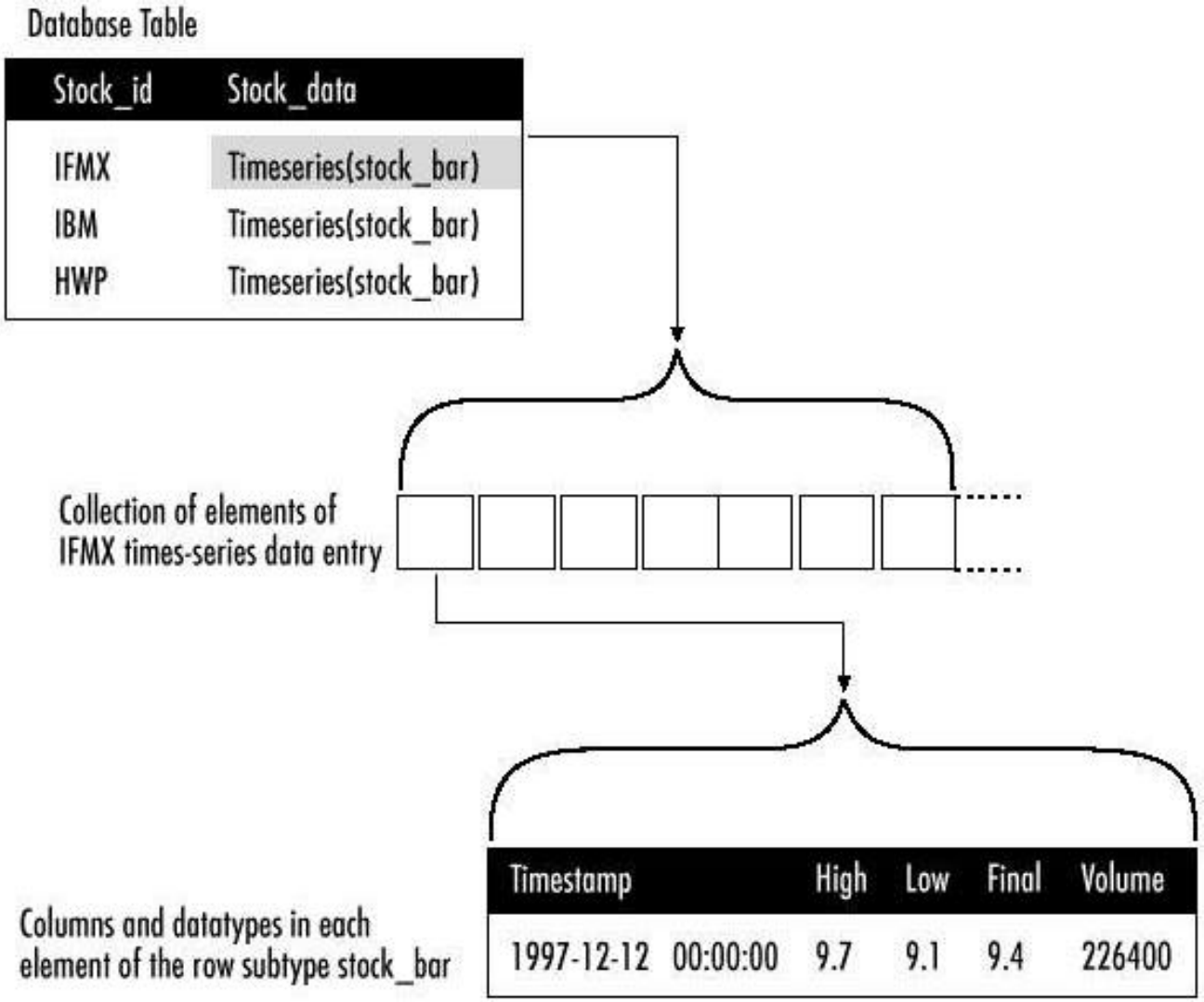
- Langage d'expression OLAP pour MS SQL Server
- Exemples
  - SELECT
    NON EMPTY {[Time].[1997], [Time].[1998]} ON COLUMNS,
    [Promotion Media].[Media Type].Members ON ROWS
    FROM Sales

  - SELECT
    {[Measures].[Unit Sales]} ON COLUMNS,
    ORDER( EXCEPT( [Promotion Media].[Media Type].members, {
    [Promotion Media].[Media Type].[No Media]}),[Measures].[Unit
    Sales], DESC) ON ROWS
    FROM Sales

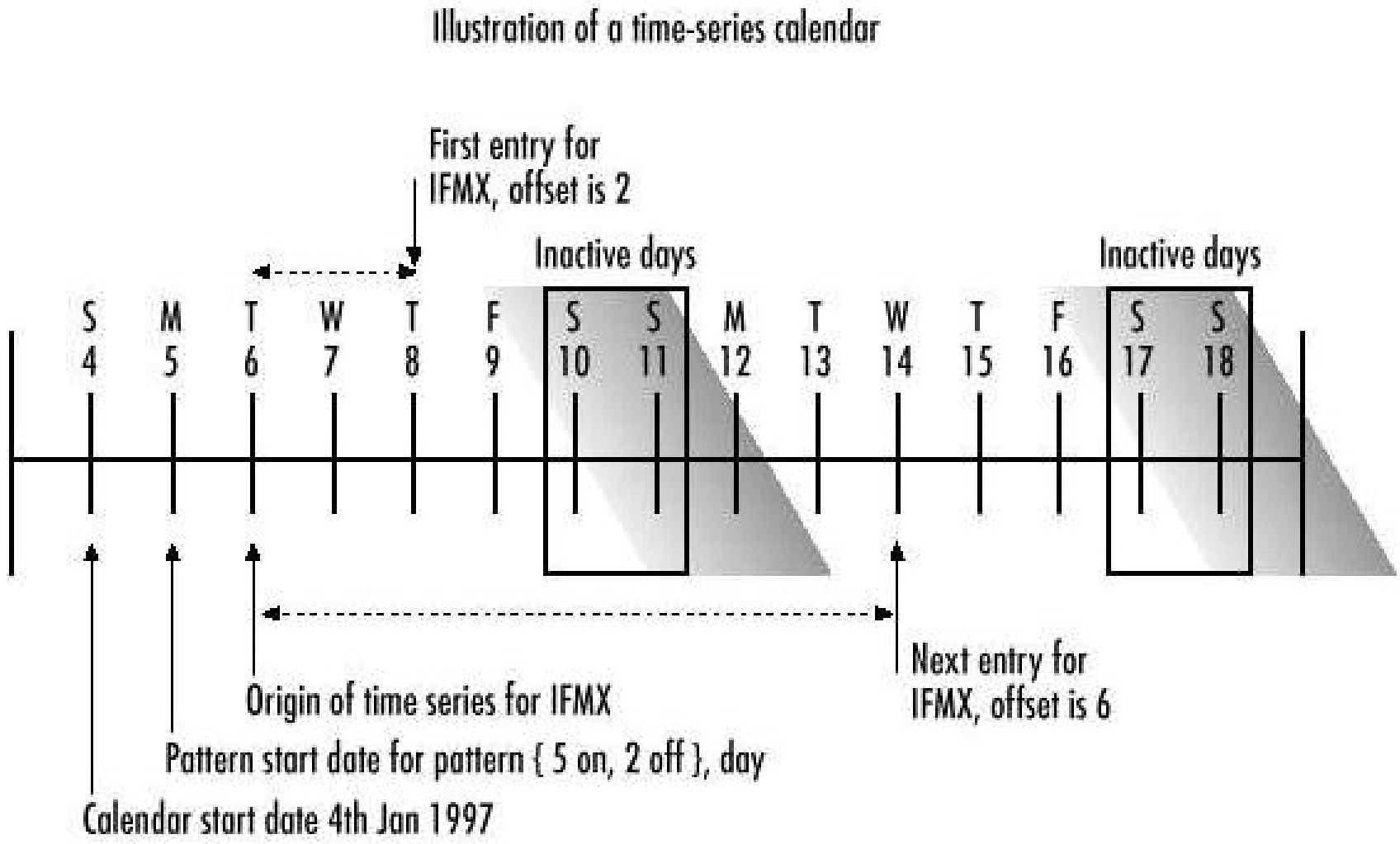
# ADT Séries Temporelles (Time Series)

- Définition
  - Suite de couple (Valeur, estampille de temps)
- Applications
  - Finance (stock value), Santé (épidémiologie), ...
- Type
  - calendar, ...
- Opérations
- Index
  - par rapport au temps

# ADT Séries Temporelles



# ADT Séries Temporelles



# Architecture des SGBD décisionnels et des infocentres

- Chargement de travail
  - Requêtes complexes (nombreuses jointures + aggrégats)
  - Très gros volumes de données
- Réponses
  - SGBD Parallèles sur machines parallèles (SMP,Cluster, …)
  - RAID et SAN (Storage Area Networks)
  - Index Bitmaps, algorithme

# Aspects Systèmes

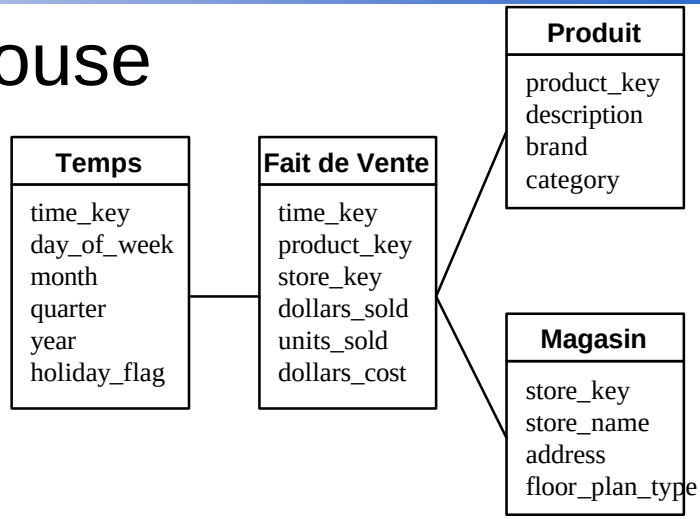
- Stockage
  - Tables de fait
    - Append only (RID ≈≈ Timekey)
    - Ligne de fait

      clés et mesures de type de taille fixe

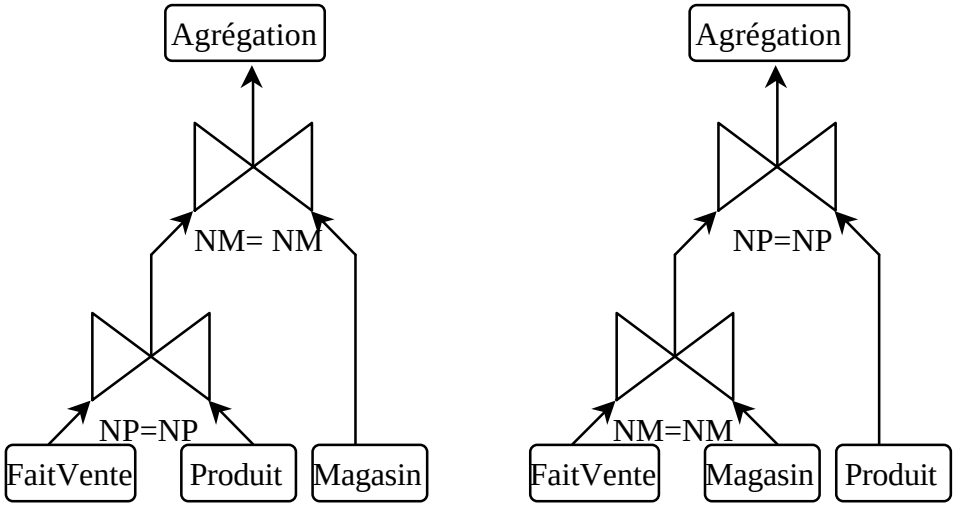
      enregistrement de taille fixe (accès aléatoire)
- Jointure en Etoiles
  - Star Query
- Index Binaire (Bitmap)

# Jointure en Etoile (Star Join)

- Exemple de Data Warehouse

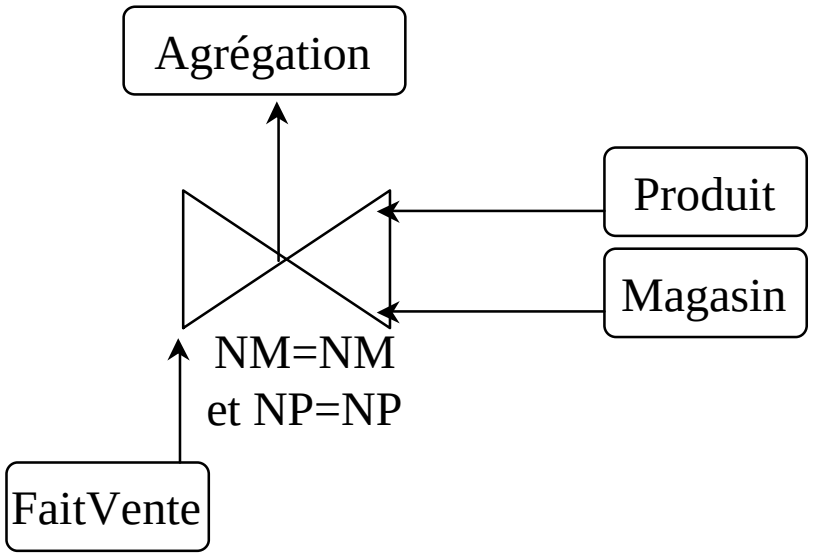



- SGBD classique
  - 2 stratégies



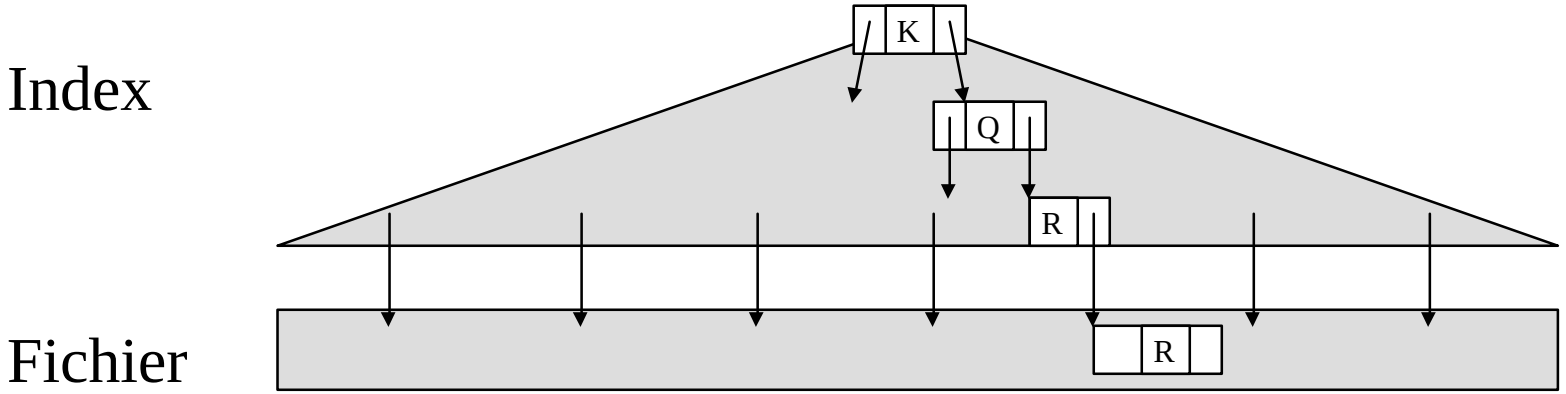
# Jointure en Etoile (Star Join)

- Opérateur de jointure n-aire
  - Star Join
    - 1 source : la table de fait
    - 1 à N sources : les tables de dimension



# Index Binaire (Bitmap)

- Rappel sur les index B-Tree
  - couple <Valeur Attribut, Pointeur d 'enregistrement>



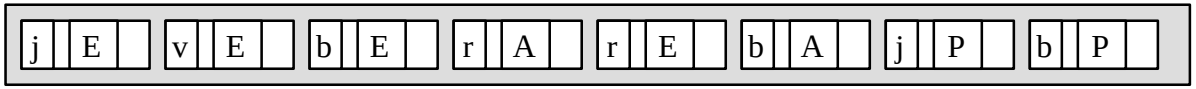
# Index Binaire (Bitmap)

- Une chaîne de bits par valeur ou groupe de Valeur ou sur un prédicat

Index des Régions

| | |
|---|---|
| (E)urope | 11101000 |
| (A)sie | 00010100 |
| (P)roche Orient | 00000011 |

Fichier




Index des Couleurs

| | |
|---|---|
| (r)ouge | 00011000 |
| (b)leu | 00100101 |

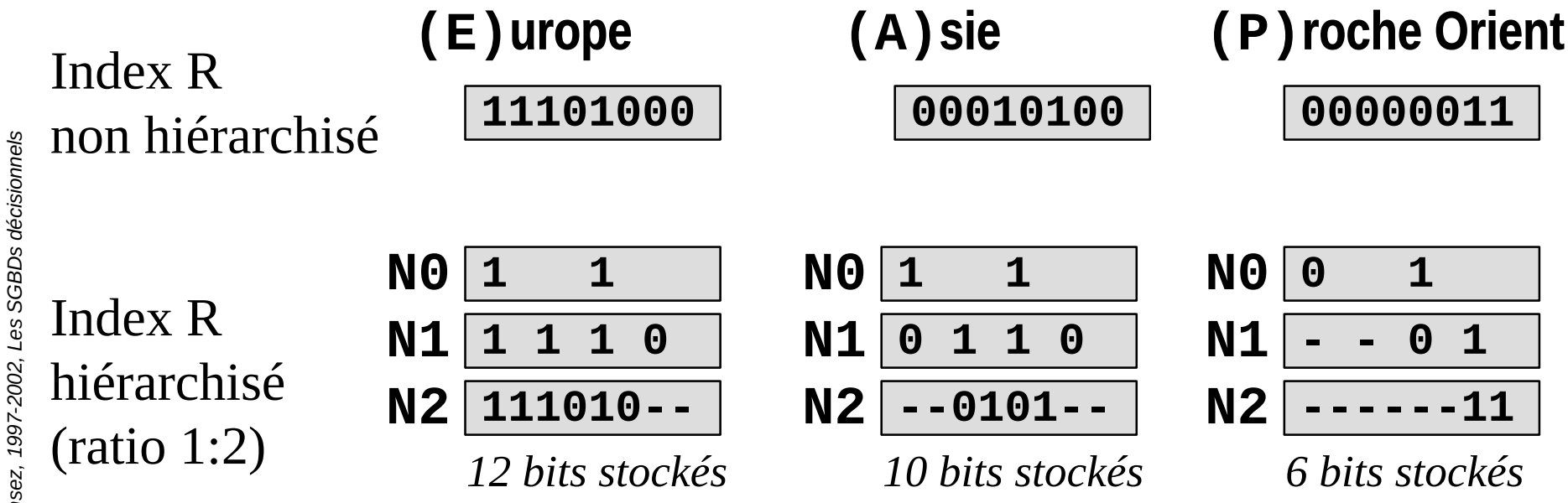
- Opérations logiques OR, AND, NOT

| | |
|---|---|
| E | 11101000 |
| A | 00010100 |
| Asie OR Europe | 11111100 |

| | |
|---|---|
| E | 11101000 |
| Not (R OR B) | 11000010 |
| Jaune AND Europe | 11000000 |

# Optimisation des Index Binaires

- Compression
  - doit permettre les comparaisons
- Hiérarchisation multi-niveau
  - ratio entre niveaux (ex 1:32)



# Index Bitmap

- Indexation de la table de fait
  - sur les valeurs des tables de dimension associées
  - Cardinalité faible des attributs de la dimension
  - Tables de dimension à faible évolution
  - Exemple
    - CREATE BITMAP INDEX sales_region_ix ON sales(region);
- Jointure en étoile (Star query)
  - 2 étapes :
  - Sélection des faits et Sélection des dimensions
    - index bitmap pour les faits
  - Jointure en étoile
- Parallélisation
  - balayages // dans les étapes 1 et 2

# Index Bitmap

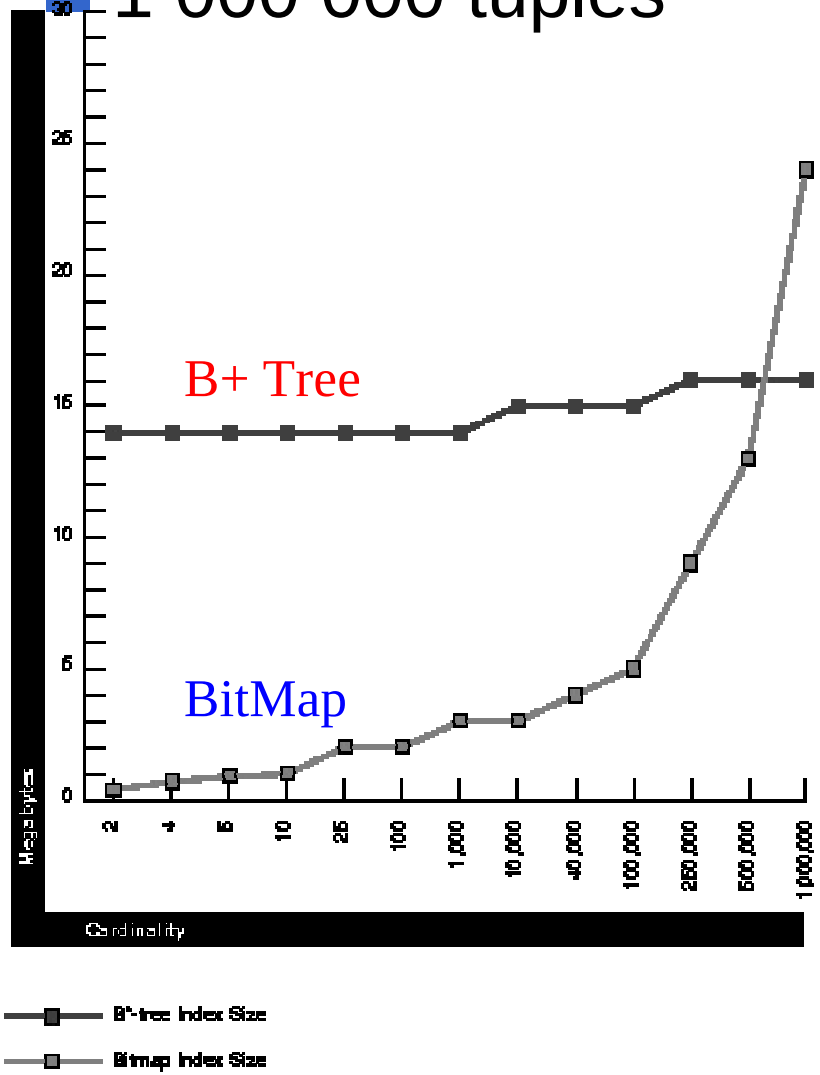
- Compression du bitmap
  - Informix Online XPS, Oracle 8, ...
- Indexation bitmap partielle
  - Informix Online XPS
- Indexation bitmap de colonnes virtuelles
  - Colonne calculée
    - Vente.Profit = Vente.Prix_Vente - Produit.Prix_Conseillé
- Bibliographie
  - "Star Query Processing in the Oracle7 Server," Oracle White Paper, March 1996.
  - O'Neil, P. E., Graefe, G., "Multiple-table joins through bitmapped join indexes," SIGMOD Record, Vol. 24, No. 3, September 1995.

# Performance des Index BitMap

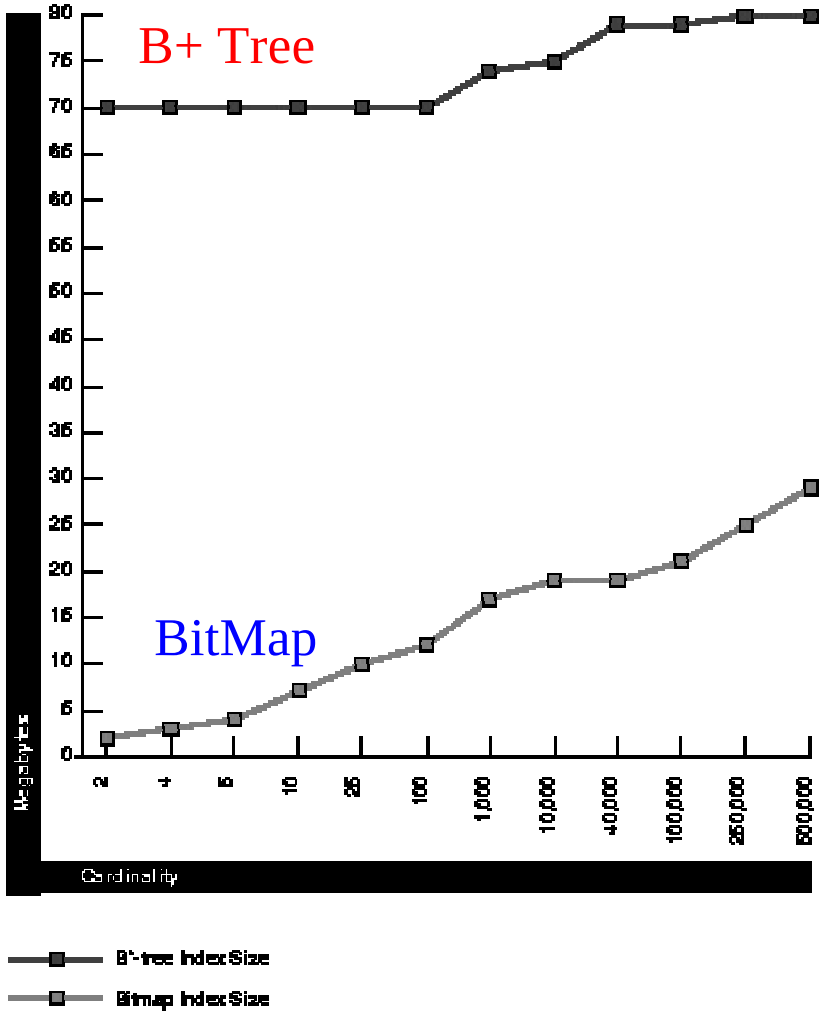
- Pour les tables Append Only
  - Éviter l'OLTP (modification des attributs)
- Domaine de cardinalité faible
  - faible nombre de valeurs différentes

# Occupation des Index BitMap

## 1 000 000 tuples




## 5 000 000 tuples



# Autres techniques

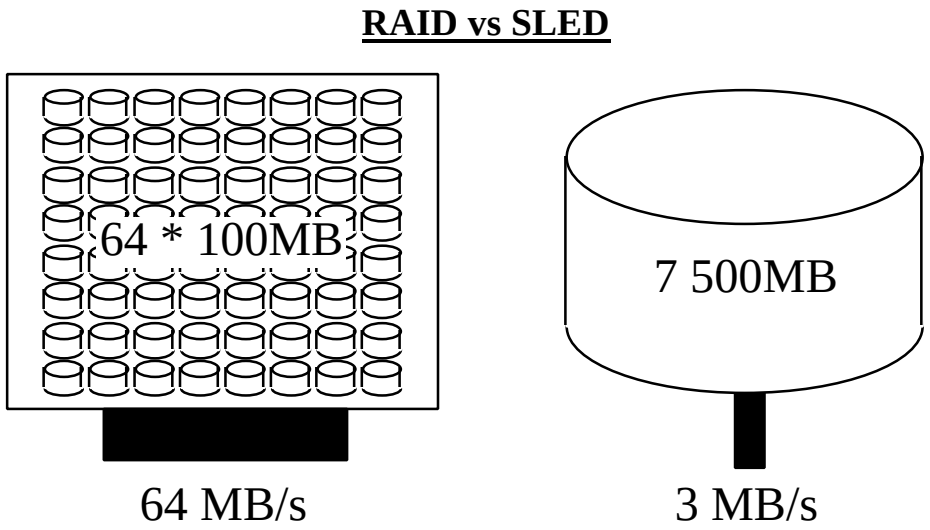
- Cache (de calcul) d'agrégats
  - Conserve et réutilise les tables temporaires issues de calculs d'agrégat précédents
- Vue matérialisée (SNAPSHOT)
  - Recalcul incrémental de la vue

# Chargement de l'entrepôt

- À partir des bases de productions
- Outils
  - Extracteurs
    - Journaux des images, Déclencheurs, Requetages, …
  - Purifieur
  - Transformateur
    - Format pivot : de + et + XML
  - Chargeur

# RAID Redundant Array of Inexpensive Disks [Patterson 88]

- En 1988
- SLED : Single Large Expensive Disk



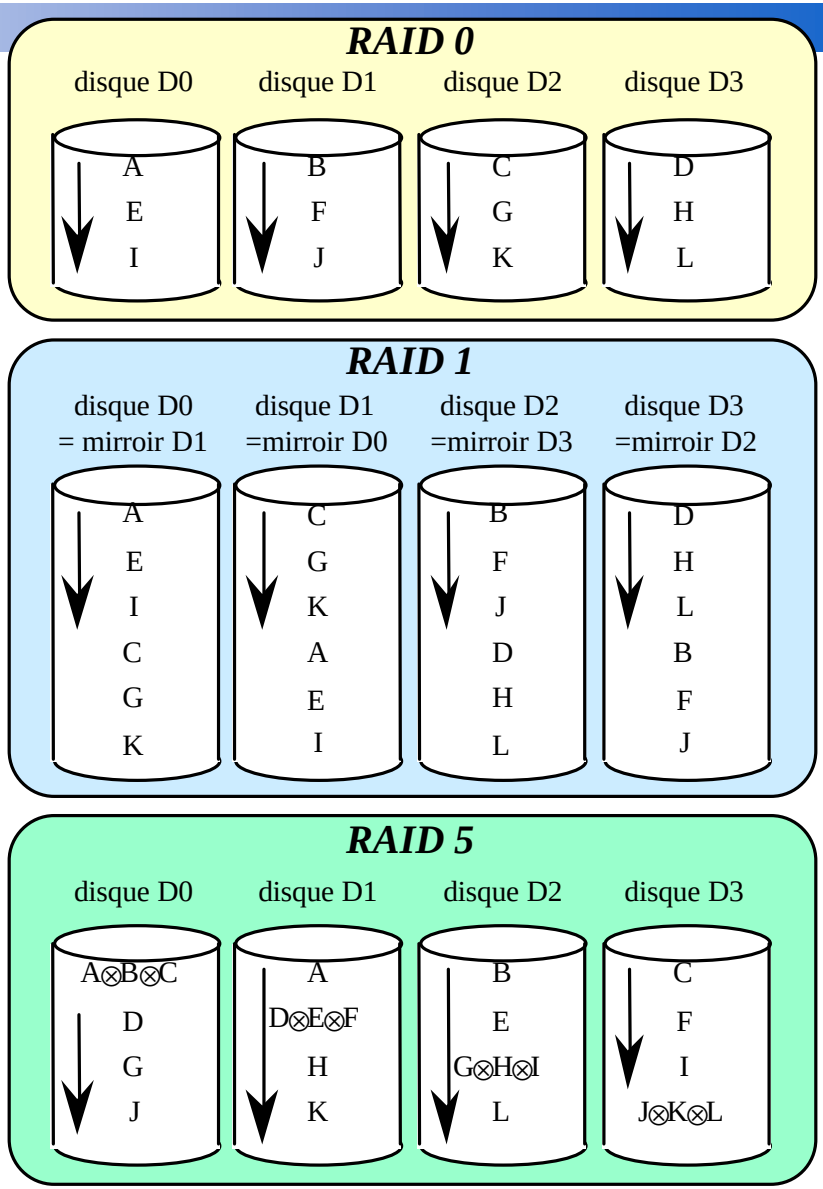

- RAID : Redundant Array of Inexpensive Disks
  - tableau de disques peu coûteux pour améliorer les débits I/O
  - cependant il faut introduit de la redondance
    à cause de la MTBF qui suit une loi de poisson

# Niveaux de fonctionnement des RAID

- Redondance
  - Niveau 1 : Les Disques Mirroirs
    - TANDEM Mirrored Disks
  - Niveau 2 : Code de Hamming pour ECC
    - CM2 DataVault
  - Niveau 3 : Un Seul Disque de Contrôle par Groupe de Disque
  - Niveau 4 : Lectures et Ecritures indépendantes
  - Niveau 5 : Contrôle réparti sur les disques du Groupe
- Sans redondance
  - Niveau 0 : Stripping
    - répartition des blocs contigus d'un fichier entre les disques mais pas de redondance (ie AID)

# Redondance et Balayage (scan) parallèle

- RAID 0
  - répartit les données sur plusieurs disques pour améliorer les performances.
- RAID 1
  - effectue des copies miroirs de disques pour survivre aux pannes.
- RAID 5
  - utilise la correction d'erreur et la répartition des données pour fournir performance et sécurité de manière économique et efficace.
  - La correction est basée sur la propriété du XOR (⊗) :
    ( X ⊗Y ) ⊗ Y = X



# Nouveaux Niveaux de fonctionnement des RAID

- Redondance
  - Niveau 0/1 ou « 10 » : Stripping sur des disques mirroirs
  - Niveau 6 : Contrôle redondant
    - Remarque

      2,3,4,5 ne resiste qu'à la panne d'un seul disque dans le groupe
    - Le niveau 6 ajoute du contrôle pour la redondance
    - Plusieurs propositions

      N+Q : autre fonction que XOR

      2D
      - » Soit une matrice de disques
      - » les lignes forment des groupes RAID5
      - » les colonne forment des groupes RAID5

# Contrôleur RAID

- Caractéristiques
  - Niveaux de RAID géré 0, 1, 5, 0/1
  - Hot Sparing : Secours permutable (disques de réserve)
  - Hot Swapping : Echange à chaud (par un opérateur humain)
- Contrôleur RAID actuel
  - 30 disques Hot Plug 1 Go par groupe de 5 pour du RAID niveau 0,1,5
  - Configuration en 1996 : des disques SCSI de 1 à 4 Go
    (organisés par groupe de 5 pour RAID 5)
- Interfaces
  - Low Cost : PCI + Disques SCSI
  - UW-SCSI
  - FiberChannel (25 Mo/s)
    pour des Architectures en Clusters

# SAN (Storage Area Network)

- Réseau de stockage et d'archivage haute performance
  - Clusters de machines
    partageant des clusters de RAID

# Standardisation

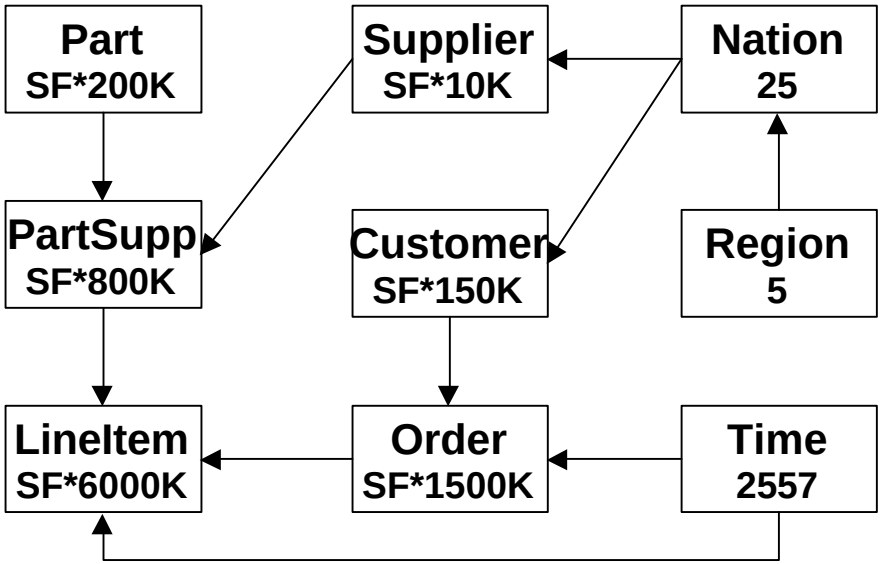
- Trop souvent propriétaire
- Standardisation du Référentiel
  - Echec de AD/Cycle et Metadata Coalition
  - OIM Open Information Model
    - MS avec TI et Platinium Technology
    - OLE for OLAP
    - PLATO: Surcouche multidimensionnelle d ’SQL Server

# Benchmarks (Banc de Performances)

- Mesurer les performances d'un système (matériel / logiciel) sous une charge de travail caractérisant une application type d'infocentre.
- Objectif
  - comparer les produits entre eux (avec d'acheter)
  - dimensionner son système en fonction de ses besoins
- Les benchmarks DW du TPC
  - http://www.tpc.org/
  - TPC-D : BD Décisionnelles (InfoCentre) *Obsolète*
  - TPC Benchmark H (TPC-H)
    *ad-hoc, decision support benchmark*
  - TPC Benchmark R (TPC-R)
    *business reporting, decision support benchmark*

# TPC-D : BD Décisionnelles (InfoCentre)

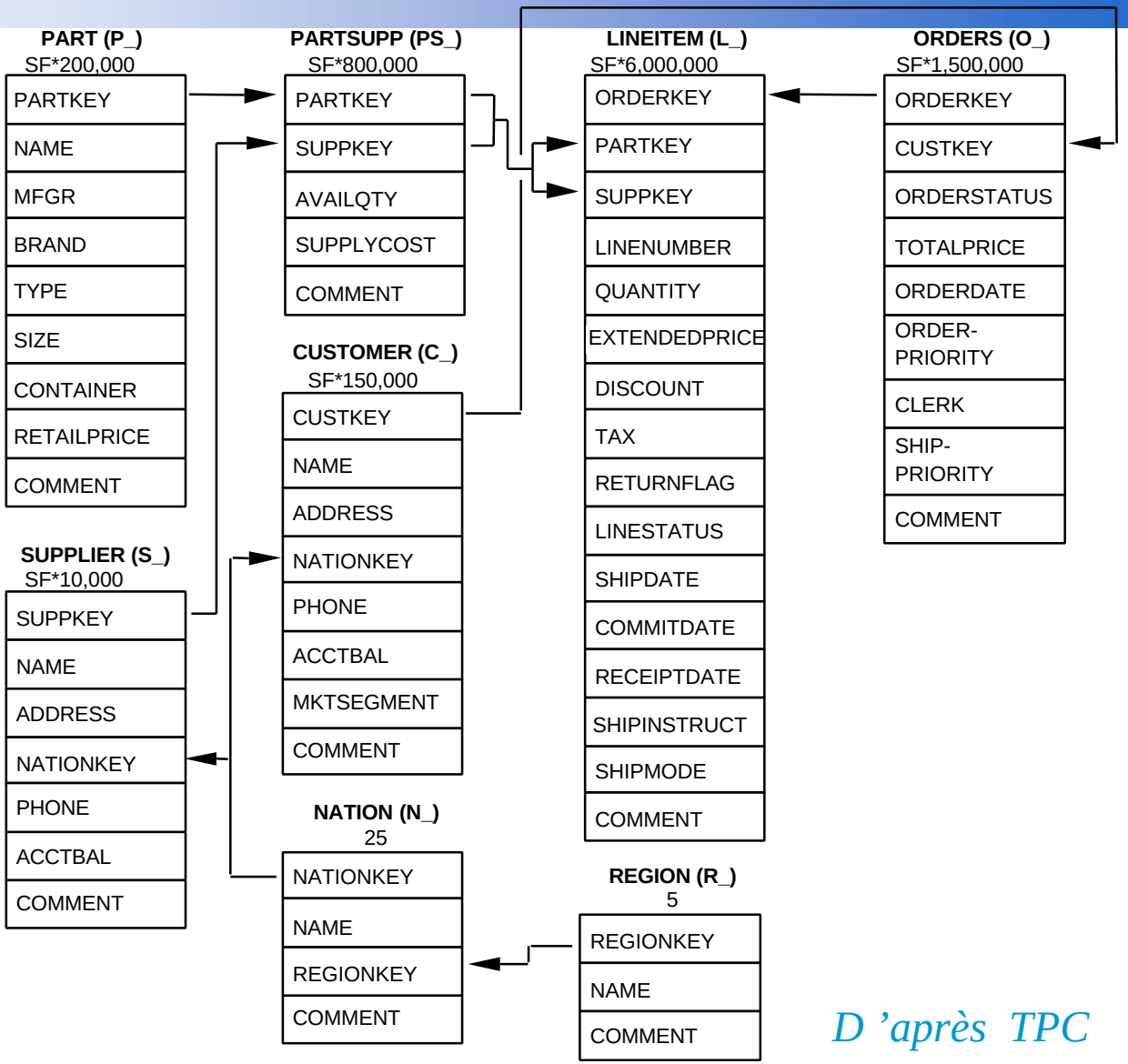
- Data Warehouse & Data Mining
- SF : Scale Factor
  - 1,10,30,100,300,1000,3000,10000
  - soient des BDs de 1 Go à 10 To
- Modèle de données du TPC-D




*D 'après [Bes97]*

# Modèle de données du TPC-D H R

- 8 tables




*D 'après TPC*

# TPC Benchmark

- decision support benchmark

## The TPC Benchmark H (TPC-H)

*ad-hoc*

- suite de requêtes ad-hoc (orienté business) et de modifications concurrentes

  large volumes of data,

  queries with high degree of complexity

- Mesure
  - TPC-H Composite Query-per-Hour Performance Metric (QphH@Size)

## The TPC Benchmark R (TPC-R)

*business reporting, decision support benchmark*

- Similaire au TPC-H, mais permet des optimisations sur les requêtes complexes.
- Mesure
  - TPC-R Composite Query-per-Hour Performance Metric (QphR@Size)

## Taille des bases

- SF (Scaling Factor)=1 (~1Go), 10, 30, 100, 300, 1.000, 3.000, 10.000

# Suite de Requêtes du TPC-H & R

- Pricing Summary Report Query (Q1)
- Minimum Cost Supplier Query (Q2)
- Shipping Priority Query (Q3)
- Order Priority Checking Query (Q4)
- Local Supplier Volume Query (Q5)
- Forecasting Revenue Change Query (Q6)
- Volume Shipping Query (Q7)
- National Market Share Query (Q8)
- Product Type Profit Measure Query (Q9)
- Returned Item Reporting Query (Q10)
- Important Stock Identification Query (Q11)
- Shipping Modes and Order Priority Query (Q12)
- Customer Distribution Query (Q13)
- Promotion Effect Query (Q14)
- Top Supplier Query (Q15)
- Parts/Supplier Relationship Query (Q16)
- Small-Quantity-Order Revenue Query (Q17)
- Large Volume Customer Query (Q18)
- Discounted Revenue Query (Q19)
- Potential Part Promotion Query (Q20)
- Suppliers Who Kept Orders Waiting Query (Q21)
- Global Sales Opportunity Query (Q22)
- New Sales Refresh Function (RF1)
- Old Sales Refresh Function (RF2)

# Exemple de résultats

**Exercice : quelle configuration choisir pour un niveau de perf =2800 QphH**

- TPC-H Results - Revision 1.X - 300GB Scale Factor
  - 01/2001

| Company | System | SF | QphH | Price Perf.($/QphH) | Total Sys. Cost | Currency | Database Software | Operating System | CPU Type | # CP | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| HP | NetServer LXr 8500 | 300 | 1402.5 | 207 | 290737 | US $ | Microsoft SQL Server 2000 | Microsoft Windows 2000 | Intel Pentium III Xeon 700MHz | 8 | N |
| Compaq | ProLiant 8000-8P | 300 | 1506.8 | 280 | 422173 | US $ | Microsoft SQL Server 2000 | Microsoft Windows 2000 | Intel Pentium III Xeon 700MHz | 8 | N |
| Compaq | AlphaServer ES40 Model 6/667 | 300 | 2832.1 | 1058 | 2995034 | US $ | Informix XPS 8.31 FD1 | Compaq Tru64 UNIX V5.1 | Alphachip 21264 667 MHz | 16 | Y |
| IBM | NUMA-Q 2000 | 300 | 4027.2 | 652 | 2625301 | US $ | IBM DB2 UDB 7.1 | DYNIX/ptx 4.5.1 | Intel Pentium III Xeon 700MHz | 32 | N |
| Compaq | AlphaServer GS320 Model 6/731 | 300 | 4951.9 | 983 | 4865968 | US $ | Informix XPS 8.30 FC3 | Compaq Tru64 UNIX V5.1 | AlphaChip 21264A 731 MHz | 32 | N |
| IBM | NUMA-Q 2000 | 300 | 5923.2 | 653 | 3868930 | US $ | IBM DB2 UDB 7.1 | DYNIX/ptx 4.5.1 | Intel Pentium III Xeon 700MHz | 48 | N |
| IBM | NUMA-Q 2000 | 300 | 7334.4 | 616 | 4516767 | US $ | IBM DB2 UDB 7.1 | DYNIX/ptx 4.5.1 | Intel Pentium III Xeon 700MHz | 64 | N |

# Bibliographie - Livre


- Rob Mattison, Data Warehousing -Strategies, Technologies and Technics, IEEE Computer Society 1996, ISBN 0-07-041034-8
- Michael J. Corey, Michael Abbey, Ian Abramson and Ben Taub, « Oracle8 Data Warehousing », Ed Mc Graw Hill, ISBN: 0-07-882511-3, 686 pages
- Chris Date, « Introduction aux Bases de Données », 7ème édition, Chapitre 21.

# Bibliographie - Articles

- CACM, «Industrial-Strength Data Warehousing » Vol. 41, No. 9 September,1998
  - http://www.acm.org/cacm/0998/0998toc.html
- SQL99 On-Line Analytical Processing (SQL/OLAP)
  - ISO/IEC 9075-1/Amd1:2001 http://www.ansi.org

# Bibliographie - Articles

- Tips OLAP de SQL Server Magazine
  - http://www.sqlmag.com
- Intelligent Enterprise.
  - http://www.intelligententerprise.com/info_centers/data_warehousing/
- Parts de marché des outils OLAP
  - http://www.olapreport.com/Market.htm