

Université Mohammed Premier  
Faculté des Sciences  
Département Informatique

## Informatique décisionnelle DatawareHouse

Master Ingénierie Informatique - Semestre 2  
Année universitaire 2020-2021

Pr Ibtissam ARRASSEN

## **Plan**

- [Section 1 : Introduction aux entrepôts de données](#)
- [Section 2 : La modélisation multidimensionnelle](#)
- [Section 3 : Le processus de conception](#)
- [Section 4 : OLAP \( Atelier 2\)](#)
- [Section 5 : Datamining \( Atelier 3\)](#)
- [Section 6 : Ateliers SQL Server 2008](#)

## Liste des ateliers de travail

Atelier 1 : **SSIS (Sql Server Integration Services)**

Exemple de projets SSIS.

Atelier 2 : **Quelques requêtes OLAP sur AdventureWorks.mdf**

Atelier 3 : **SSAS (Sql Server Analysis Services)**

Atelier 4 : **SSAS : Datamining / les règles d'association**

Atelier 5 : **SSAS : les arbres de décision**

Atelier 6: **SSRS- Sql Server Reporting Services**

## Introduction

- [Problématique](#)
- [Applications](#)
- [Les Entrepôts de données](#)
- [Différences entre SI et SID](#)
- [Les problèmes liés à l'entreposage de données](#)
- [Architecture d'un système décisionnels](#)

### Problématique

- **Objectif**

Améliorer les performances décisionnelles de l'entreprise

- Décisions stratégiques
- Décisions rapides

### Problématique

Les entreprises passent à l'ère de l'information.

Défi :

Transformer leur système d'information qui avait une vocation **de production** à un SI décisionnel dont la vocation **de pilotage** devient majeure.

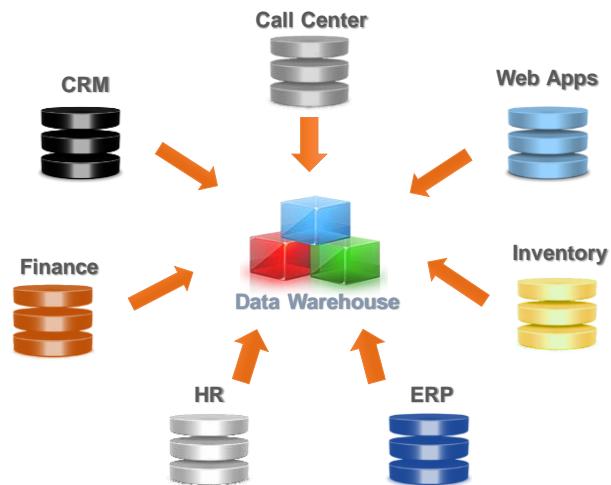
### Problématique

- **Comment ?**
- Répondant aux demandes d'analyse des décideurs :
  - Qui sont mes clients ?
  - Qui sont mes meilleurs clients actuellement?
  - Pourquoi sont-ils mes clients?
  - Comment les conserver ou les faire revenir ?
  - Ces clients seront-ils intéressants pour moi ?
  - Quels sont nos 10 produits les plus bénéficiaires sur la période 2018-2019?

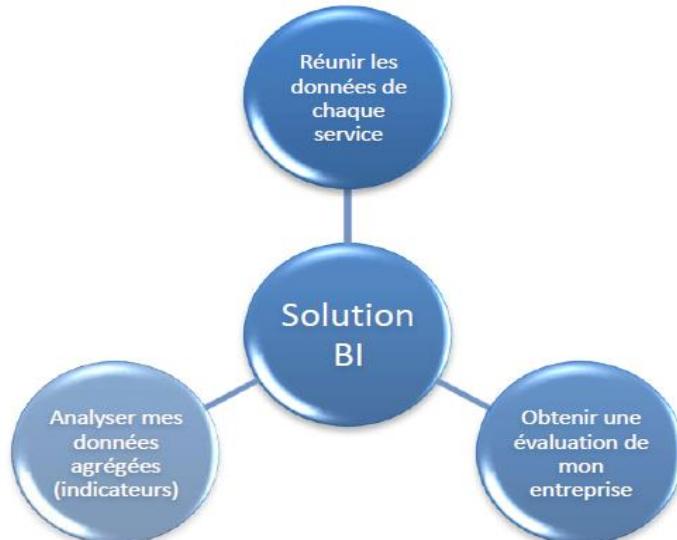
### Problématique

- Problèmes à prendre en compte
  - **Une grande masse de données :**
    - Distribuée
    - Hétérogène
    - Très Détailée
  - **A traiter :**
    - Synthétiser / Résumer
    - Visualiser
    - Analyser
  - **Pour une utilisation par :**
    - des experts et des analystes d'un métier
    - NON informaticiens
    - NON statisticiens

### Enterprise Data Source Structure



### Solution BI (Business Intelligence)



Les jeunes entreprises se soucie d'abord d'avoir un grand nombre de clients

Les entreprises plus anciennes cherchent à fidéliser les clients et leur proposer d'autres produits.

=> C'est ce que l'on appelle **Customer Relationship Management** (CRM ou gestion des relations clients).

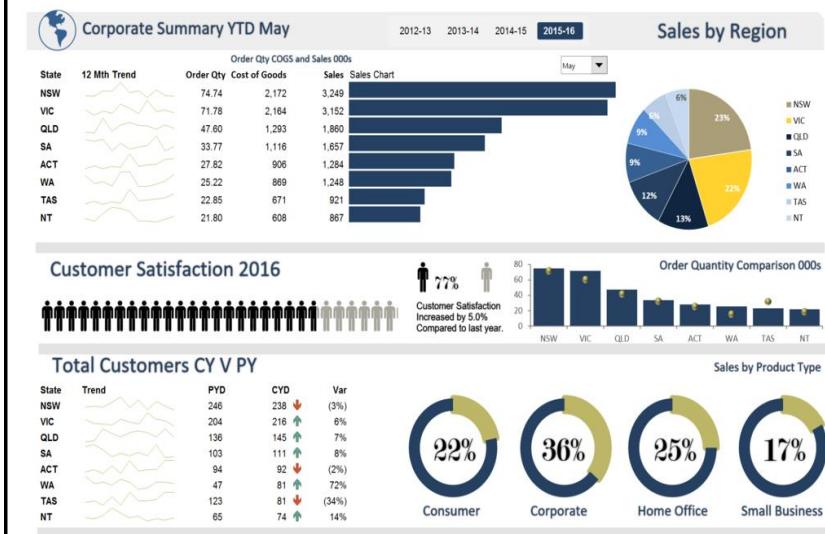
### Domaines d'Applications

- Banque, Assurance
  - Déterminer les profils client(bons ou mauvais clients).
  - Risque d'un Prêt, Prime plus précise.
- Commerce
  - ciblage de clientèle
  - déterminer les promotions (périodes, produits).
  - aménagement des rayons (2 produits en corrélation).

## Exemples de requêtes

- Quels sont les 5 produits les plus vendus pour chaque sous-catégorie de produits qui représente plus de 20% des ventes dans sa catégorie de produits ?
- Quelle est la priorité d'expédition et quel est le revenu brut potentiel des commandes de livres qui ont les 10 plus grandes recettes brutes parmi les commandes qui n'avaient pas encore été expédiées ?

## Visualisation des Données



### Qu'est ce qu'un Datawarehouse?

- Un grand réservoir de **données détaillées et sommaires** qui décrit l'organisation et ses activités.
- Il est organisé de manière à **faciliter la récupération de l'information** décrivant les activités de l'entreprise.

### Les Entrepôts de données

Un entrepôt de données est une collection de données

- orientées sujets
- intégrées,
- variables dans le temps et
- non volatiles,

en soutien au processus de **prise de décisions** de gestion.

## Les Entrepôts de données

Un entrepôt de données est une collection de données

- Orientées Sujets:

l' entrepôt est structuré autour des principaux **sujets** de l'entreprise (tels que les clients, les produits et les ventes) au lieu des principaux domaines d'application (comme la facturation client, la gestion des stocks et la vente des produits...).

Ceci reflète le besoin de disposer de données de support à la décision et non plus simplement de données orientées application.

## Les Entrepôts de données

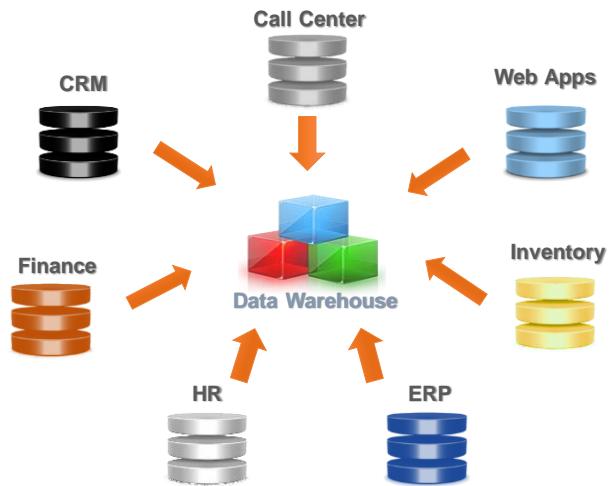
Un entrepôt de données est une collection de données

- Intégrées,

Les données sources sont souvent incohérentes, puisqu'elles arborent des formats différents.

Ces sources de données doivent subir une mise en cohérence pour présenter une vue unifiée des données aux utilisateurs.

### Enterprise Data Source Structure



### Les Entrepôts de données

- Variables dans le temps

La variation temporelle de l'entrepôt de données est montrée par :

1. la période étendue pendant laquelle les données sont conservées,
2. l'association implicite ou explicite du temps à toutes les données,
3. le fait que les données représentent une suite d'instantanés.

## Les Entrepôts de données

- Non volatiles

Les nouvelles données viennent s'ajouter en supplément à la BD et non en remplacement.

La BD absorbe continuellement ces nouvelles données en les intégrant de manière incrémentielle

=>

il n'y a pas de modification mais bien des ajouts aux données précédentes.

## Les Entrepôts de données

Le but ultime de l'entreposage de donnée:

- Intégrer les données provenant de toutes les sources de l'entreprise en un seul annuaire.
- Trouver des réponses à des requêtes, générer des rapports et effectuer des analyses.

Un ED est une technologie d'analyse et de gestion de données.

## Bénéfices de l'entreposage de données.

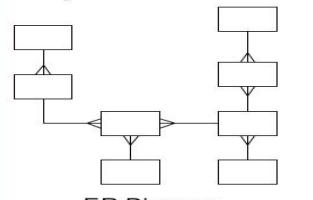
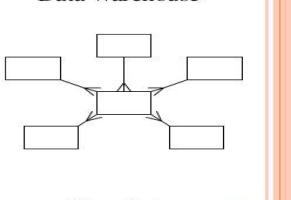
- Un taux de rendement du capital investi potentiellement élevé.
- Un avantage concurrentiel.
- Une productivité accrue de la part des décideurs d'entreprises.

## utilisateur

OLTP	OLAP
 Système d'entreprise	 Système d'entreprise
Beaucoup d'utilisateurs <ul style="list-style-type: none"> <li>• Un seul compte à la fois</li> <li>• Une seule vision métier</li> <li>• Exécutent un grand nombre de fois la même tâche</li> <li>• Lisent et modifient les données</li> <li>• Le système de données est vivant et opérationnel</li> </ul>	<ul style="list-style-type: none"> <li>• Peu d'utilisateur</li> <li>• Traitent plusieurs compte</li> <li>• On ne demande pas deux fois la même chose</li> <li>• Lisent les données récapitulées</li> <li>• pas de mise à jour des données, et agrégation de ces données.</li> </ul>

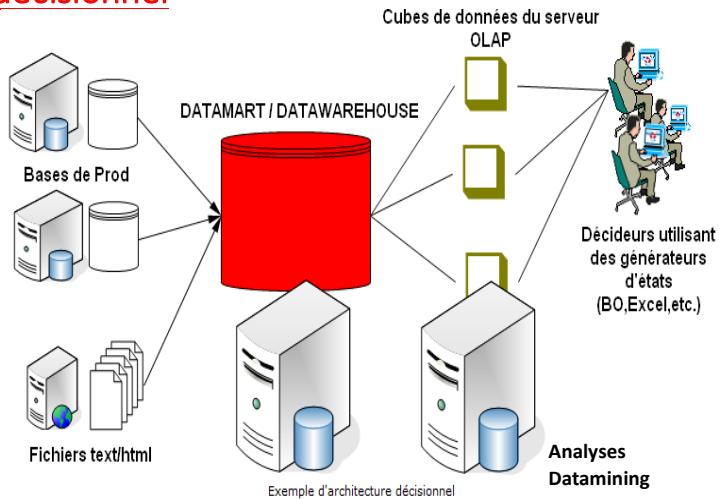
<b>Les données</b>	
<b>OLTP</b>	<b>OLAP</b>
Données orientées <b>application</b> .	Données orientées <b>sujet</b> .
Nécessaire au fonctionnement de l'entreprise.	Analyser le fonctionnement de l'entreprise.
<b>Normalisé</b> (3ème forme normale).	<b>Agrégées</b> et résumées (le détail intéresse peu).
Données courantes non historisées	Le temps est important.
Données opérationnelles: provenant des sources elles-mêmes.	les données de l'OLAP proviennent des différents OLTP.
les données changent constamment.	les données sont stables et synchronisées dans le temps.
Donnent une photographie du processus business en cours.	Donnent une vue multidimensionnelle de différents types de busines.

<b>Traitements</b>	
<b>OLTP</b>	<b>OLAP</b>
Les opérations sont des requêtes insertions et mises à jour <b>courtes et rapides</b> .	Le rafraîchissement des données est <b>périodique</b> et relativement <b>long</b> à exécuter.
Les requêtes sont <b>standards</b> et <b>simples</b> , retournent un petit ensemble de tuples	requêtes <b>complexes</b> nécessitant des Agrégations. <b>assistées, pré planifiées</b> , navigation aléatoire (drill-down).
L'ensemble des données / transaction accédés est évalué d'une 10 aine à une 100 aine.	Les batch de rafraîchissement et les requêtes complexes peuvent s'exécuter sur <b>plusieurs heures</b> .
Considération des règles de gestion.	Considération des règles de gestion métier, stratégies globales et sources de données.

Structure de la BD	
DESIGN DIFFERENCES	
<b>Operational DBMS</b>	
	
<b>Beaucoup de tables</b> <b>Une 100aines MB/GB</b>	<b>Peu de tables mais de grande taille 100aines de GB/TB</b>
<b>Petites requêtes sur une seule table</b>	<b>Requêtes larges sur une grosse quantité de données</b>
<b>Schéma hautement normalisé</b>	<b>Schéma dénormalisé schémas en étoile ou en flocon de neige.</b>

Administration du système	
OLTP	OLAP
Forte disponibilité du système	« Faible » disponibilité du système
Sauvegardes fréquentes et périodique	Sauvegardes peu fréquentes mais très volumineuses
Beaucoup de petites transactions	une transaction par jour/semaine....(chargement de données).
Peu de maintenance offline	Beaucoup de maintenance offline

## Architecture d'un Système d'information décisionnel



## Processus d'alimentation d'un ED

comment alimenter l'entrepôt ?

- Faut-il ramener toutes les données sous le même format ?
- Si oui, quel format choisir et pourquoi ?
- Sinon, comment faire pour interroger toutes ces différentes structures ?
- Quel(s) langage(s) d'interrogation va-t-on utiliser ?
- Quelle architecture utiliser ?

→ problématique de l'ETL (Extracting Transforming and Loading)

## Opérations

- **Extraction (Extraction) :**

filtrer les données à partir de données sources (BD, fichiers, sites web...) dans des BD temporaires.

- **Transformation (Transformation) :**

Transformer les données extraites dans un format uniforme.

- **Chargement (Load) :**

charger les données transformées dans la BD cible. La BD cible est souvent implantée avec un SGBD relationnel-objet.

- **Agrégat et Groupement (Aggregating and Grouping) :**

stocker les données opérationnelles et les données issues de calculs.

## Transformation

- Résolution des problèmes survenant lors de l'intégration des schémas

- Demande une solide connaissance de la sémantique des schémas

- Peu traitée par les produits du marché

- Nombreux travaux de recherche

=> Opération généralement réalisée à la main...

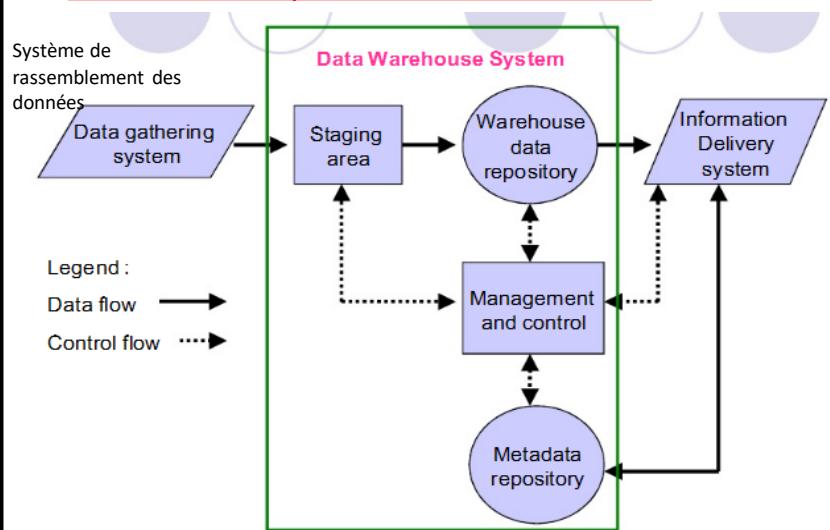
## Chargement des données

- Objectif : Stockage des données nettoyées et préparées dans le magasin de données.
- Opération :
  - ✓ risquant d'être assez longue
  - ✓ plutôt mécanique
  - ✓ la moins complexe

Mais il est nécessaire de définir et mettre en place :

- des stratégies pour assurer de bonnes conditions à sa réalisation
- une politique de rafraîchissement

## Modèle d'un système Datawarehouse



### Staging area ( Zone de transit)

les données sont prêtes à être déplacé dans :

- le référentiel d'ED
- le référentiel de Métadonnées

### Les métadonnées

Toutes les informations nécessaires pour la construction et l'administration de l'entrepôt

#### **Informations présentes dans l'entrepôt**

- données source
- données dérivées,
- dimensions,
- hiérarchies
- contraintes d'intégrités
- schéma de l'entrepôt
- indexes, partitions
- requêtes prédefinies ...

#### **Informations d'administration**

- règles de nettoyage,
- transformation,
- extraction
- politique de rafraîchissement
- sécurité
- monitoring,
- statistiques
- traçage des données

## La modélisation dimensionnelle

[Concepts de base](#)

[Tables de dimensions](#)

[Cubes](#)

[Tables de faits](#)

[Hiérarchies](#)

[Schéma en étoile](#)

[Schéma en flocon](#)

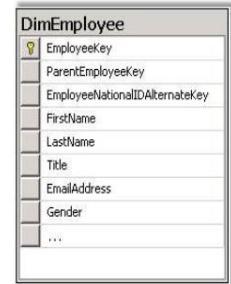
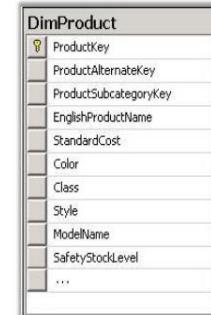
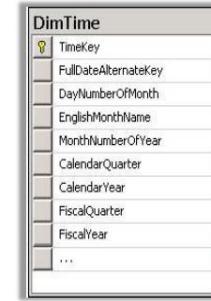
[Avantages de la modélisation multidimensionnelle](#)

[Mini-entrepôts de données \(data marts\)](#)

### Concepts de base

#### Table de dimension

- Décrit les objets du métier, comme : **employee, product, customer, service.**
- La seule dimension partagée par tous les processus est appellée “conformed” dimension.



## Concepts de base

### Fact table ( table de fait)

- Chaque fait contient les mesures associées à un processus métier spécifique.
- Un **enregistrement** dans la table de faits est une mesure, et un événement mesure produit toujours un enregistrement dans la table de faits.
- Ces événements ont souvent des **mesures numériques** qui quantifient l'ampleur de l'événement, telle que la quantité commandée, montant des ventes, durée d'appel.
- Ces nombres sont appellés faits (ou measures *dans* Analysis Services).
- La **clé** pour la table de fait est une clé composée d'un sous-ensemble des clés étrangères de chaque table de dimension impliquée dans le fait.

## Concepts de base

### Fact table ( table de fait) : Exemple

FactResellerSales	
ProductKey	
OrderDateKey	
ResellerKey	
EmployeeKey	
SalesTerritoryKey	
OrderQuantity	
TotalProductCost	
SalesAmount	

## Concepts de base

### Cube

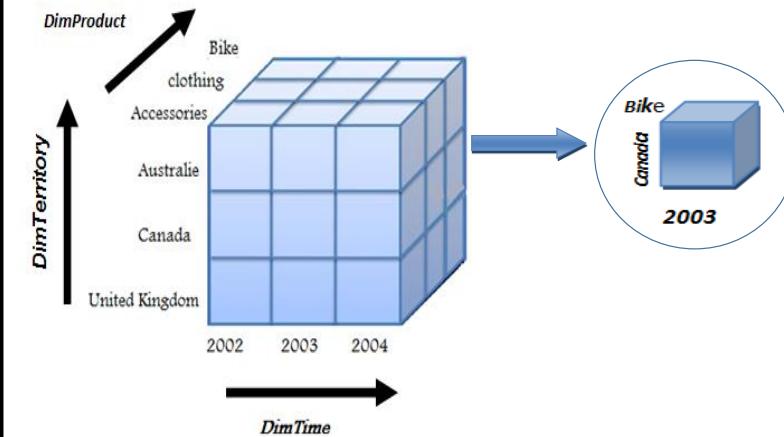
- L'unité de base pour stocker les résultats des analyses
- collection de données agrégées pour permettre de retourner rapidement les données

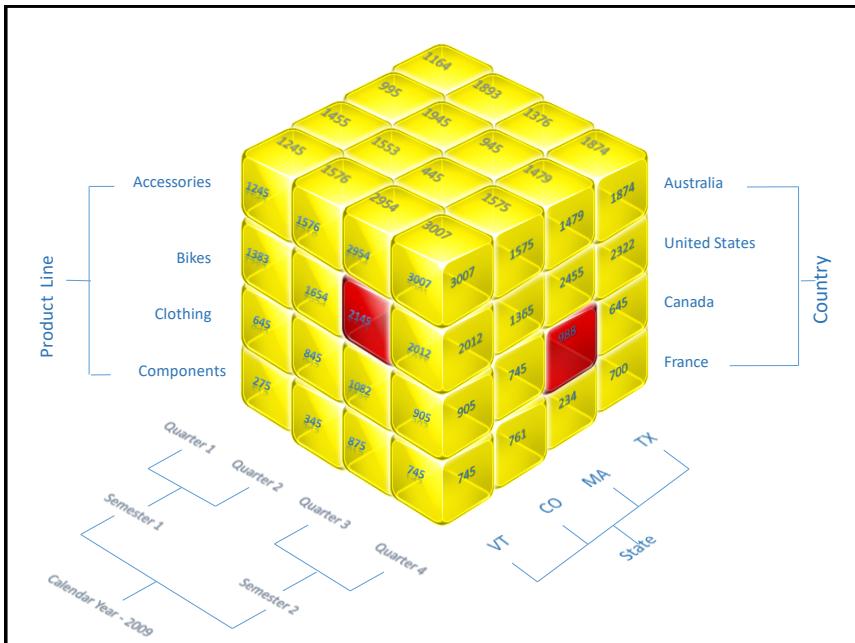
### Measure (mesure)

- Chaque cube va contenir une ou plusieurs *mesures*
- chacune basée sur une colonne dans la table de Fait qu'on veut analyser.

Exemple : ventes unitaires, profit...

## Exemple de Cube

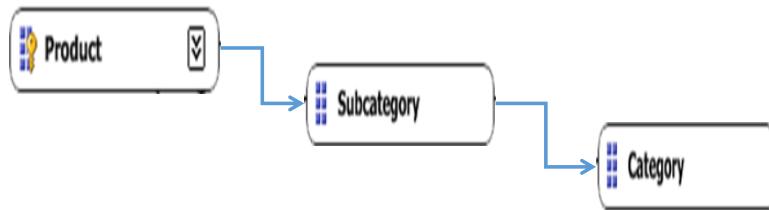




## Concepts de base

### Hierarchy (hiérarchie)

- Une hiérarchie est une collection de niveaux logiquement structurés en fonction des attributs. Dans certaines hiérarchies, chaque membre attribut unique implique l'attribut de membre au-dessus.



**Product**

- Category
- Large Photo
- Model Name
- Product
- Product Line
- Style
- Subcategory
- Product Categories
  - Members
    - All Products
    - Accessories
    - Bikes
      - Mountain Bikes
      - Road Bikes
      - Touring Bikes
    - Clothing
    - Components
  - Category
  - Subcategory
  - Product
  - Product Model Lines
- Promotion
- Reseller

**Date.Calendar**

CY 2004

**Country**

United States

Category	Subcategory	Product	Internet Sales Amount	Reseller Sales Amount
Accessories			\$407,050.25	\$76,027.18
Bikes	Mountain Bikes	\$3,814,691.06	\$2,539,198.92	
	Road Bikes	Road-250 Black, 44	\$136,827.60	\$290,269.98
		Road-250 Black, 48	\$156,374.40	\$250,687.71
		Road-250 Black, 52	\$141,714.30	\$172,989.18
		Road-250 Black, 58	\$146,601.00	\$168,591.15
		Road-250 Red, 58	\$153,931.05	\$159,795.09
		Road-350-W Yellow, 40	\$267,055.43	\$361,719.61
		Road-350-W Yellow, 42	\$261,952.46	\$227,914.29
		Road-350-W Yellow, 44	\$248,344.54	\$104,100.59
		Road-350-W Yellow, 48	\$261,952.46	\$420,082.10
		Road-550-W Yellow, 38	\$131,097.33	\$151,938.44
		Road-550-W Yellow, 40	\$142,302.23	\$120,340.63
		Road-550-W Yellow, 42	\$157,989.09	\$90,759.69
		Road-550-W Yellow, 44	\$132,217.82	\$75,296.93
		Road-550-W Yellow, 48	\$128,856.35	\$149,249.27
		Road-750 Black, 44	\$112,317.92	\$34,019.37
		Road-750 Black, 48	\$109,617.97	\$115,236.57
		Road-750 Black, 52	\$126,897.65	\$94,606.25
		Road-750 Black, 58	\$104,218.07	\$55,402.97
		Total	\$2,920,267.67	\$3,042,999.81
	Touring Bikes		\$2,427,366.12	\$2,369,136.82
	Total		\$9,162,324.85	\$7,951,335.55

## Product Hierarchies

**Bikes**

**Mountain Bikes**

**Mountain-100**

Top-of-the-line competition mountain bike. Performance-enhancing options include innovative HL Frame, smooth front suspension for all terrain.

**Road Bikes**

**Road-150**

This bike is ridden by race winners. Developed with the Adventure Works Cycles preference.

**Touring Bikes**

**Touring-1000**

Travel in style and comfort. Design emphasizes safety and performance. Wide gear range takes you on all hills. High-tech aluminum alloy construction provides durability without added weight.

Product Number	Product Name	Color	Size	Weight	Dealer	List Price
BK-M825-48	Road-150	Blue	60	25.90	\$1481.94	\$2384.07
BK-M825-44	Road-150	Blue	60	25.90	\$1481.94	\$2384.07
BK-R93R-56	Touring-1000	Blue	60	25.90	\$1481.94	\$2384.07
BK-T79U-60	Touring-1000	Blue	60	25.90	\$1481.94	\$2384.07

## Concepts de base

### Level (niveau)

- Chaque couche dans la hiérarchie est appellée niveau.

### Exemple

- “week level” ou “month level” dans l’hiérarchie “fiscal time”,
- “city level” ou a “country level” in a “geography” hierarchy.

## Concepts de base

### Schéma

- Les “Fact tables” et les “dimension tables” sont reliées. les relations dans le cube forment **un schema**.
- Deux schémas basiques : star (étoile) and snowflake (flocon de neige).

### Exemple,

- Table de fait : Ventes,
- Tables de dimensions : Clients, Produits, Temps...  
Ces tables sont reliées et forment un schema..

## La modélisation dimensionnelle

Une technique de conception logique ayant pour but de présenter les données dans une forme standard, intuitive, qui permet des accès à grandes performances.

Un modèle dimensionnel

=

Table de faits (clé composite)

+

Tables de dimensions ( clé simple )

## Clés de substitution

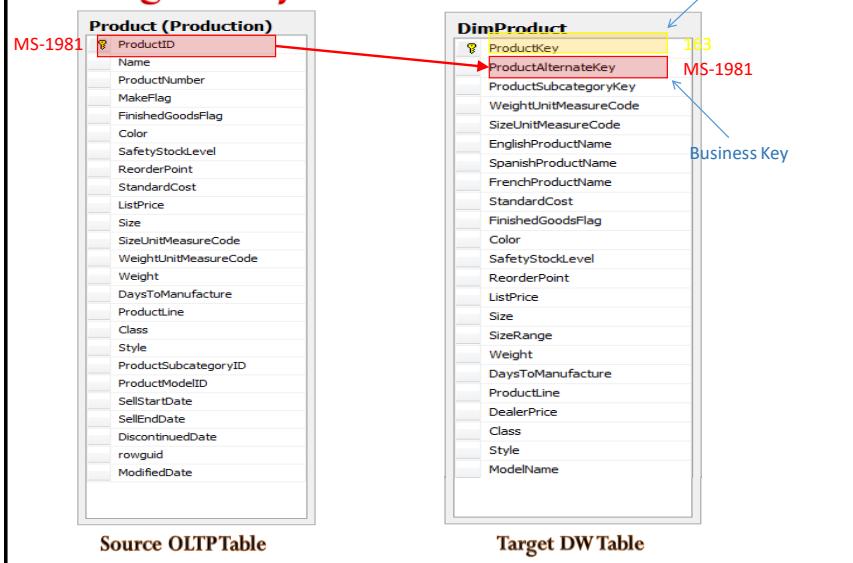
Tt les clés naturelles sont remplacées par des clés de substitution.

Tt **jointure** entre Tables de Faits et Tables de dimensions se base sur des **clés de substitution** et non des clés naturelles

=>

les **données de ED conservent une indépendance** par rapport aux données produites par les SI classiques.

## Surrogate Keys : clés de substitution



## La modélisation dimensionnelle

### Schéma en étoile (Star Schema)

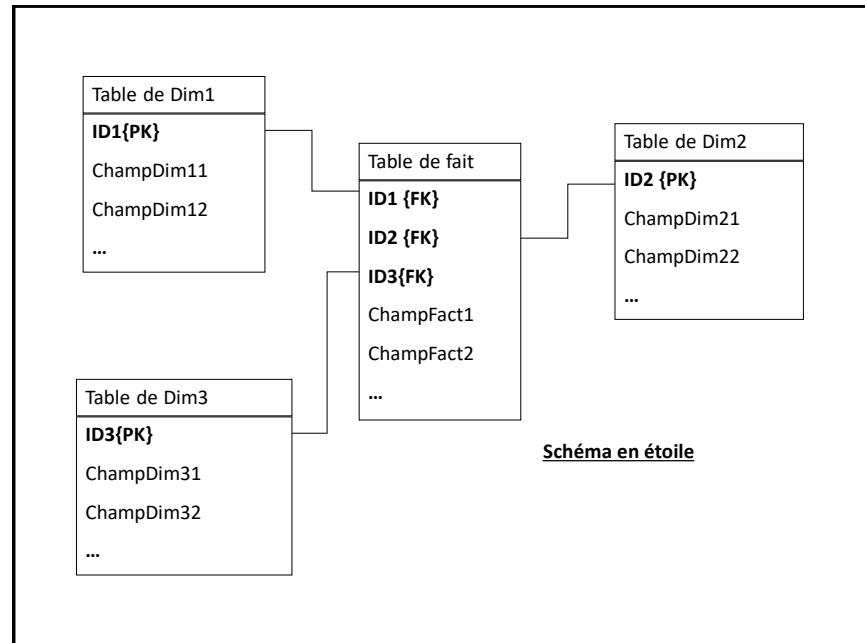
Une structure logique constituée d'une table de faits, contenant des données factuelles au centre et entourée par des tables de dimensions qui contiennent des données de référence (éventuellement dénormalisées).

## La modélisation dimensionnelle

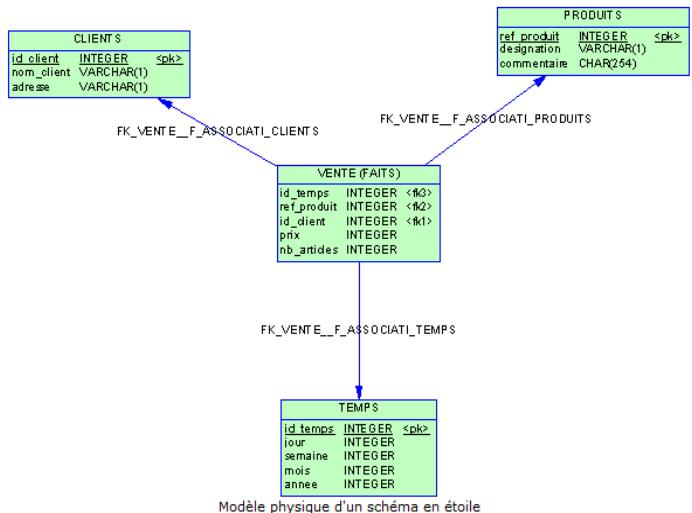
### Schéma en étoile

Permettent d'augmenter les performances des requêtes grâce à la dénormalisation des informations de référence en une seule table de dimension.

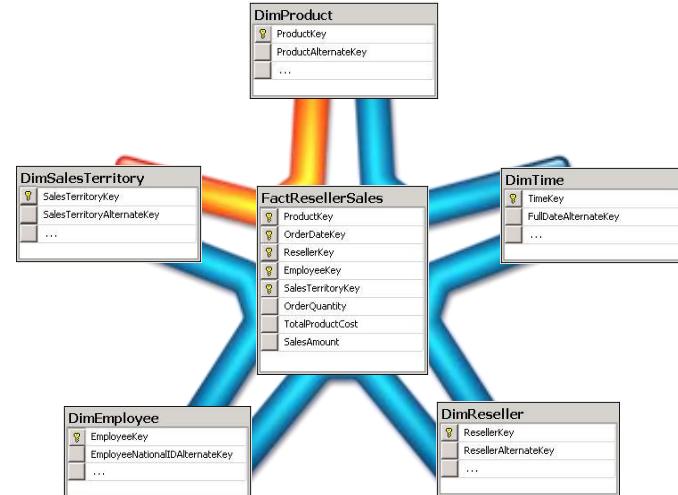
La dénormalisation s'avère adéquate lorsque la table de dimension reçoit des accès fréquents ce qui réduit la charge de jointure des tables supplémentaires pour accéder à ses attributs.



## Schéma en étoile : Exemple 1



## Schéma en étoile : Exemple 2

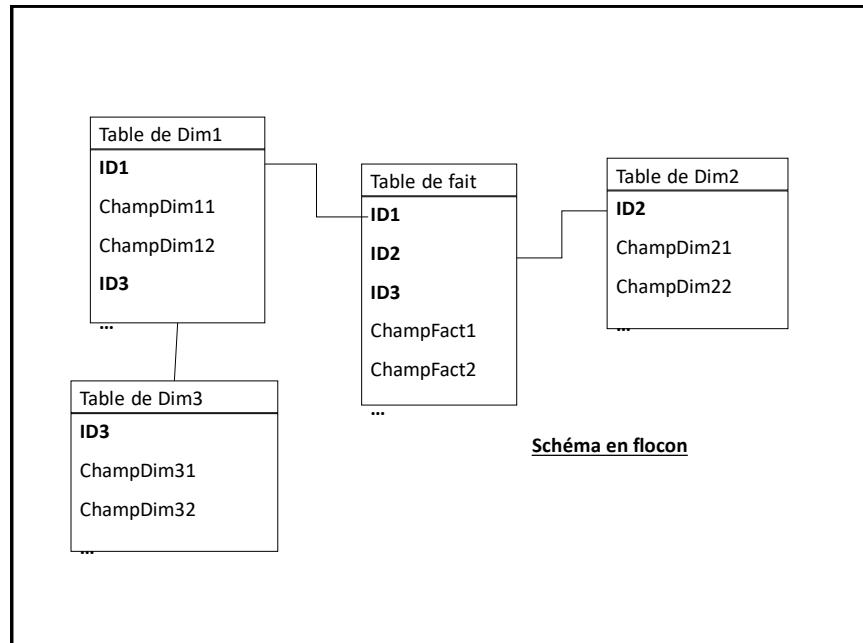


## La modélisation dimensionnelle

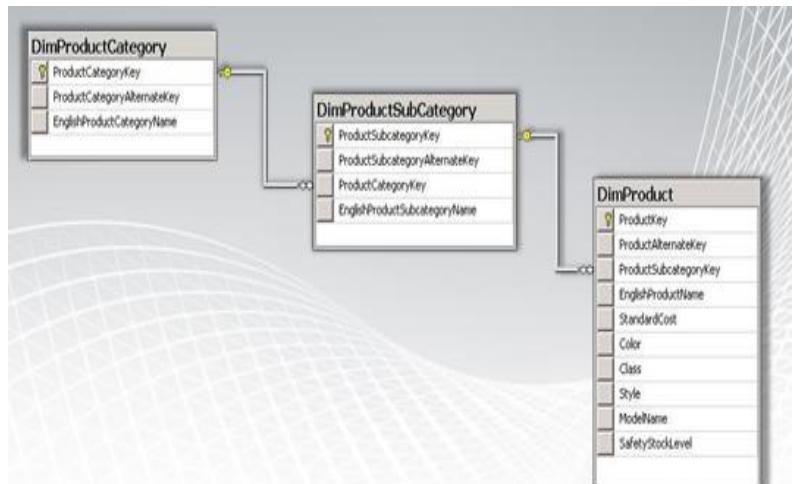
### **•Schéma en flocon (snowflake)**

Une variante du schéma en étoile, où les tables de dimensions ne contiennent pas de données dénormalisées.

Càd permettre à des dimensions d'avoir elles-mêmes des dimensions.



## Schéma en flocon : Exemple 1



## La modélisation dimensionnelle

### Schéma en étoile–flocon

- Les schémas de bases de données les plus appropriés font appel à un mélange de schémas dénormalisés en étoile et normalisés en flocon.
- Les dimensions hiérarchiques.

### Exemple:

la dimension temps qui concerne des années, peut être « décomposée » en semestre, puis en trimestre...

## La modélisation dimensionnelle

- Avantages

- L'**efficacité** des accès aux données (grâce à la cohérence de la base de données).
- La **capacité à gérer des exigences changeantes** de l'utilisateur ( schéma en étoile)
- l'**extensibilité**:
  - Ajout de nouveaux faits
  - Ajout de nouvelles dimensions
  - Ajout de nouveaux attributs dimensionnels

## La modélisation dimensionnelle

- Avantages

- La capacité à modéliser des situations professionnels communes:  
Un nombre croissant d'approches standard existent pour gérer les situations de modélisation communes dans le monde des affaires.
- Le traitement des requêtes est prévisible:  
Chaque table de faits devrait être interrogée de façon indépendante.

### Mini-entre�ots de données (data marts)

- Un datamart est un sous-ensemble d'un entre�ot de données qui satisfait les exigences d'un département, service ou d'une fonction professionnelle dans une organisation.
- Les capacités de mini-entre�ots de données atteignent plusieurs centaines de GOs et assurent des analyses sophistiquées de type, OLAP ou Datamining.
- Un mini-entre�ot contient un grand nombre de tables de synthèse et de valeurs agrégées. Ceci augmente largement la durée de la procédure de chargement.

### Mini-entre�ots de données (data marts)

- la réduction du volume des données accédés par les utilisateurs finaux => Améliorer les délais de réponse.
- Minimiser Les coûts de mise en place des entre�ots de données.  
  
Les utilisateurs d'un mini-entre�ot sont mieux définis, leurs besoins sont mieux ciblés  
  
=> une coopération plus facile avec ces utilisateurs.

## Processus de conception

- [Les éléments d'une bonne conception](#)
- [Processus de conception : 9 étapes](#)
- [Dimensions à évolution lentes](#)
- [Fait additif, semi additifs, non additifs](#)
- [Dimension Temps](#)
- [Grandes Dimension](#)
- [Dimensions Dégénérées](#)
- [Estimation de la taille de l'entrepôt de données](#)
- [Gestion de projet Data Warehouse](#)

### Les éléments d'une bonne conception

- l'entrepôt de données permet de faire **toutes les opérations analytiques** et donnera aux décideurs des moyens chiffrés pour évaluer les faits voulus.
- les dimensions seront **orientées entreprise** et pas fonction, avoir le plus possible des dimensions génériques et réutilisables.
- Les intitulés utilisés pour la création des datamarts, des dimensions, des attributs et des faits sont vraisemblablement ceux dont l'utilisateur final aura connaissance. **Choisissez-les avec soin.**
- Un attribut ne peut exister que dans une et **une seule dimension**, alors qu'un fait peut figurer dans **plusieurs tables de faits**.

## Les éléments d'une bonne conception

- Si une dimension semble apparaître à plusieurs endroits, cela signifie qu'elle joue plusieurs rôles. **Nommez ces rôles et traitez les comme des dimensions différentes.**
- **Documenter**, documenter, documenter. Un ED non documenté est un entrepôt qu'on ne peut pas faire évoluer, comprendre ou modifier. Gare à la rétention d'information !!
- N'oubliez pas, pendant votre phase d'analyse, de **lister les outputs et les questionnements** des analystes et décideurs de votre entreprise. Ceux-ci serviront de fil conducteur tout au long de votre projet.

## Les éléments d'une bonne conception

Il est fondamental de se rappeler que l'entrepôt s'alimente par le système OLTP.

- Il faut **disposer de la source de données** pour chaque attribut ou fait du modèle plus les données nécessaires à leurs consolidations
- C'est au moment de la modélisation qu'il faut s'assurer de la **disponibilité de ces données**

### Processus de conception : 9 étapes

#### Etudes de cas : Maison de rêves

1. Choisir le processus d'activité à modéliser
2. Choisir le grain du processus d'activité.
3. Identifier les dimensions et s'y conformer.
4. Choisir les faits.
5. Emmagasiner les calculs préliminaires dans la table des faits
6. Finaliser les tables de dimensions
7. Choisir la durée de la base de données
8. Suivre les dimensions à modification lente
9. Décider des priorités de requêtes et des modes de requêtes

### Processus de conception

#### **1. Choisir le processus d'activité à modéliser**

- Le sujet de l'ED . le plus important au point de vue commercial, qui est susceptible d'être livré à temps. en respectant les budgets, et est destiné à répondre aux questions professionnelles
- Le meilleur choix de premier mini-entre�t de données est souvent celui qui a trait aux « ventes ». Cette source de données est censée  tre accessible et de grande qualit .

## Processus de conception

### 1. Choisir le processus d'activité à modéliser

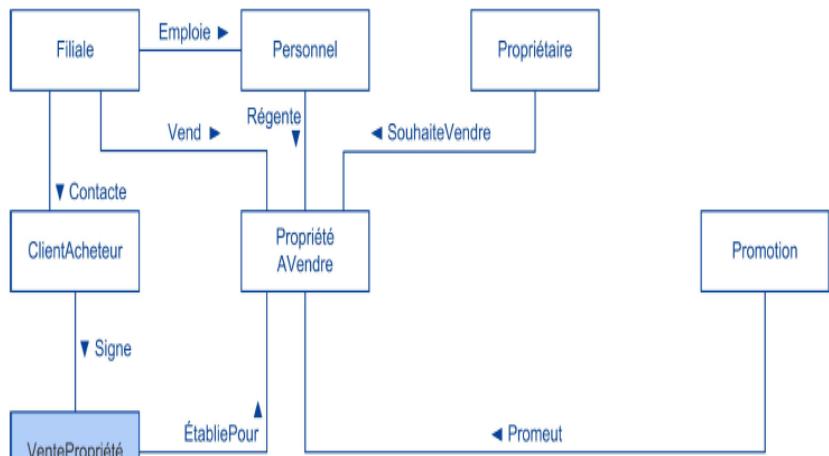
- Etude de cas : BD « Maisons de Rêve »,

les processus métier discrets de Maisons de Rêve sont :

- Les ventes de propriétés;
- Les locations de propriétés;
- Les visites de propriétés;
- La publicité des propriétés;
- L'entretien des propriétés.

## Processus de conception

- Etude de cas : BD « Maisons de Rêve »,



## Processus de conception

### 2. Choisir le grain du processus d'activité.

- décider exactement de ce que représente un enregistrement d'une table de faits.

#### Par exemple:

- l'entité **VentePropriété** représente les faits relatifs à chaque vente de propriété et devient la table des faits du schéma en étoile des ventes de propriétés .
- le grain de la table de faits **VentePropriété** est une vente de propriété individuelle.

## Processus de conception

### 3. Identifier les dimensions et s'y conformer.

- avec suffisance de détails, pour décrire des objets tel que « **clients** » « **propriétés** » avec la granularité correcte.

#### Par exemple:

- Tout client **ClientAcheteur** est décrit par les attributs (**IDClient**, **numClient**, **nomClient**, **typeClient**, **ville**, **province**, **pays**).
- Un ensemble de dimensions mal présenté ou incomplet réduira à coup sûr l'utilité de l'ED.

## Processus de conception

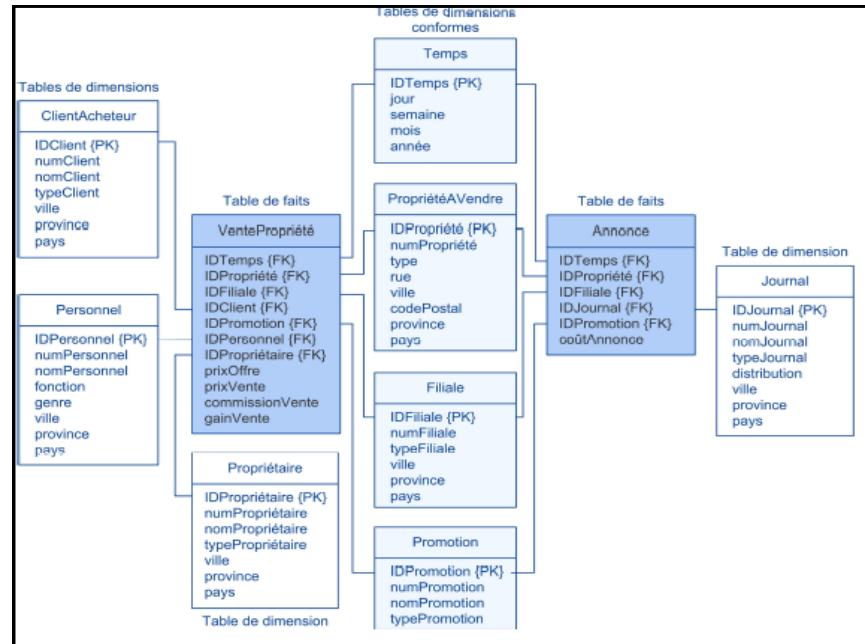
### 3. Identifier les dimensions et s'y conformer.

- une dimension peut être partagée par plus d'un mini-ED, la dimension est dite **conforme**.

- Dans « Maisons de rêves » les dimensions conformes entre ED « **Ventes** » et l'ED « **Annonce** » sont:

**Temps,**  
**PropriétéAVendre,**  
**Filiale**  
**Promotion.**

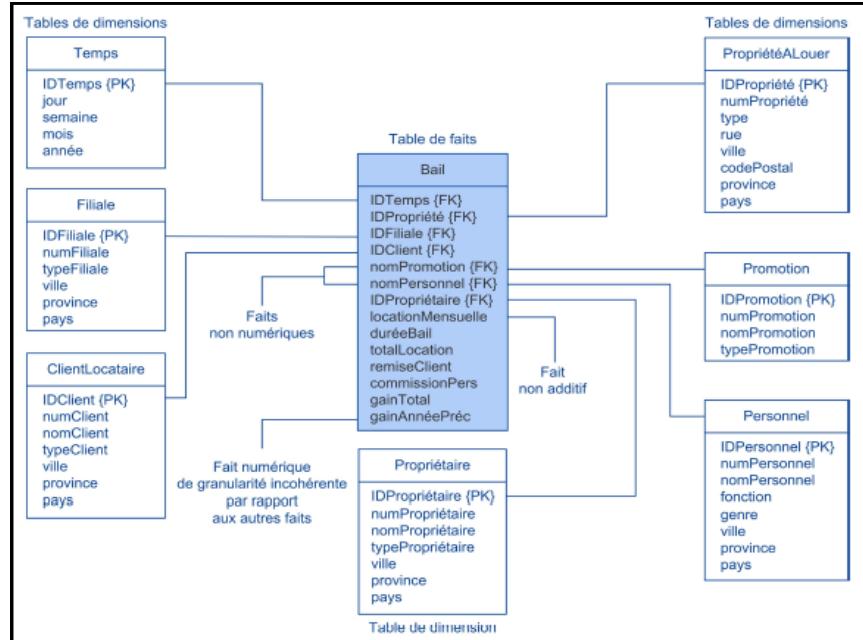
- Assurer la synchronisation de ces dimension sinon échec de l'exploitation des 2 mini ED

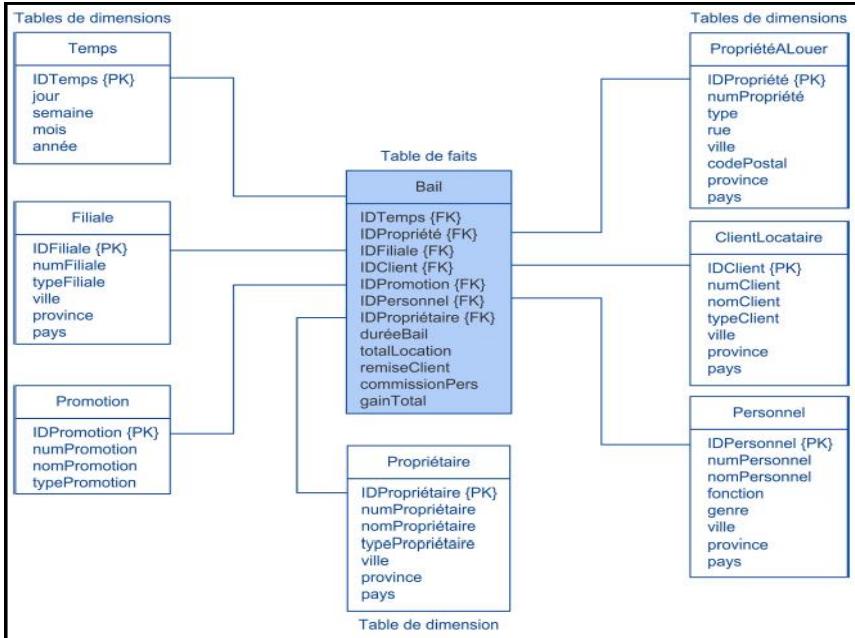


## Processus de conception

### 4. Choisir les faits (mesures).

- Le grain de la table de faits détermine les **faits utilisables dans le mini-ED**.
- si (grain = vente de propriété ) ➔ tous les faits (mesures) numériques doivent se référer à cette vente déterminée.
- Les faits doivent être, outre **numériques**, également **additifs**.





## Processus de conception

### 5. Emmagasiner les calculs préliminaires dans la table des faits

- L'exemple classique de la nécessité de mémoriser des calculs préliminaires lorsque le fait est un **profit ou une perte**.
- une valeur fondamentale pour une entreprise
- s'il y a le moindre risque qu'un utilisateur fait des calculs incorrectes

#### Exemple

**gainTotal = totalLocation - (remiseClient + commissionPers)**

=> mémoriser le **gainTotal**

## Processus de conception

### 6. Finaliser les tables de dimensions

- revenir aux tables de dimensions et y ajouter toutes les **descriptions textuelles** possibles.
- Les descriptions textuelles seront aussi intuitives et compréhensibles que possible pour les utilisateurs.
- L'utilité d'un mini-ED est en effet déterminée aussi par la portée et la nature des attributs des tables de dimensions.

## Processus de conception

### 7. Choisir la durée de la base de données

- La durée mesure le **saut dans le passé** qu'une table de faits permet d'effectuer.
- Dans certain cas on exige de parcourir des périodes d'une ou deux années précédentes.
- Dans d'autres sociétés, telles que les compagnies d'assurance, des exigences légales imposent parfois de conserver des données vieilles de cinq ans, voire plus.

## Processus de conception (SCD)

### 8. Suivre les dimensions à modification lente (SCD)

- Chaque dimension est indépendante de toutes les autres, mais parfois le contenu d'une dimension peut **changer par rapport au temps**.
- Le concepteur, doit admettre que **les dimensions sont constantes** pour conserver une structure dimensionnelle indépendante.
- Cependant, il est parfois nécessaire de procéder à des ajouts mineurs afin de se rendre compte du caractère évolutif de cette dimension.
- Ces dimensions **quasi constantes** sont appelées "dimensions à évolution lente" » « **SCD** : Slowly Changing Dimensions »

## Dimensions à évolution lentes (SCD)

### Changement de description des membres dans les dimensions

- un client peut changer d'adresse, se marier, ...
- un produit peut changer de noms, de formulations
  - « Tree's » en « M&M », « Raider » en « Twix »,
  - « Yaourt à la vanille » en « Yaourt en saveur Vanille », « bio » en « Activia »

### Choix entre 3 solutions

- écrasement de l'ancienne valeur
- versionnement
- valeur d'origine / valeur courante

### Remarque

quand la transition n'est pas immédiate : il reste pendant un certain temps des anciens produits en rayon

➔ Solution : 2 membres différents

## Dimensions à évolution lentes (SCD)

### Dimension à évolution lente du premier type : SCD1

Les anciennes valeurs sont **écrasées**. La nouvelle valeur remplace simplement l'ancienne.

Temps	#T:JJ:MM:AA:Event
1201:14:02:99:St Valentin	

Fait de Vente	#T:#C:#P:Prix
200:100:77:100	
201:100:77:100	
202:100:77:100	
202:100:66:10000	

Client	#C: #V:Nom:SitMarital
100:Didier:Divorcé:Marié	

Produit	#P:Descr
66:Bague	568:100:77:20
77:Fleur	1115:100:77:100

1116:100:77:100
1117:100:66:50000
1200:100:77:100

## Dimensions à évolution lentes (SCD)

### Dimension à évolution lente du second type : SCD2

Versionnement : On **conserve la situation initiale ainsi que la situation actuelle**. On créer un second enregistrement de la dimension avec la nouvelle valeur. Ce deuxième tuple comporte une nouvelle clé.

Temps	#T:JJ:MM:AA:Event
1201:14:02:99:St Valentin	

Produit	#P:Descr
66:Bague	568:100:2:77:20
77:Fleur	1115:100:3:77:100

Fait de Vente	#T:#C:#P:Prix
200:100:1:77:100	
201:100:1:77:100	
202:100:1:77:100	
202:100:1:66:10000	

Client	#C: #V:Nom:SitMarital:DateEffet
100:1:Didier:Célibataire:10	
100:2:Didier:Marie:203	
100:3:Didier:Divorcé:567	
100:4:Didier:Marié:1118	

## Dimensions à évolution lentes (SCD)

### Dimension à évolution lente du troisième type : SCD3

Valeur d'origine / valeur courante : Ajout de nouveaux champs pour l'attribut concerné. Il est aussi possible **d'ajouter un champ "Date d'effet"**. La valeur du nouvel attribut est écrasée et la valeur du champ (Date d'effet) est mise à jour. Le contenu du champ d'origine n'est jamais modifié.

Temps	#T:JJ:MM:AA:Event
1201:14:02:99:St Valentin	

Fait de Vente	#T:#C:#P:Prix
200:100:77:100	
201:100:77:100	
202:100:77:100	
202:100:66:10000	
568:100:77:20	
1115:100:77:100	
1116:100:77:100	
1117:100:66:50000	
1200:100:77:100	

Client	#C: #V:Nom:SMcour:SMorig:DateEffect
100:Didier:Marié:Célibataire:1118	
102:Paul:Célibataire:NULL:NULL	

## Dimensions à évolution lentes (SCD)

Une tentative de normalisation plus forte (floconage) ne résout pas les évolutions des dimensions.

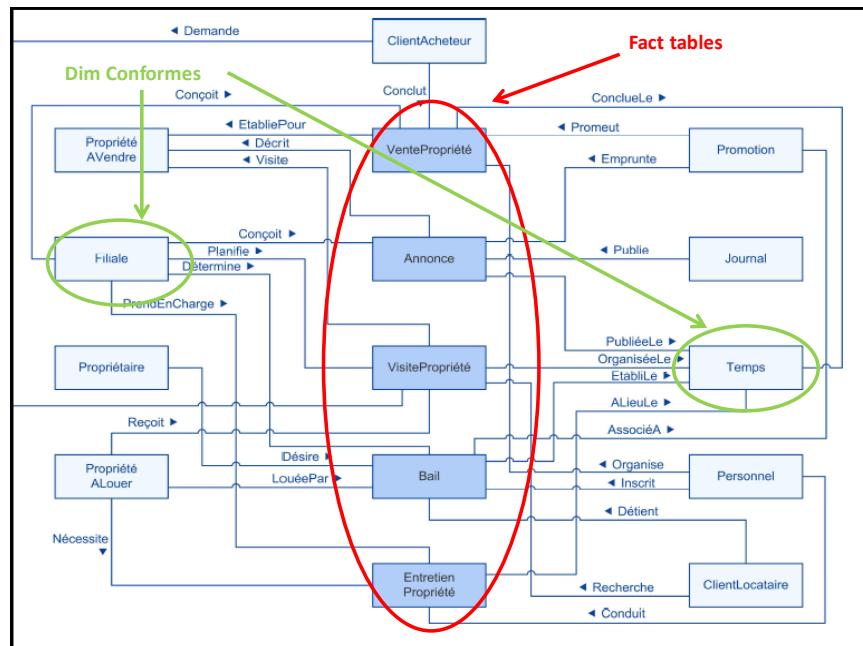
- Perte de l'interrogation par voie logicielle

La gestion de l'évolution des dimensions n'est pas une tâche facile!!!!

## Processus de conception

### 9. Décider des priorités de requêtes et des modes de requêtes

- Les soucis les plus proéminents, relatifs au design physique et qui affectent la perception du mini-ED par l'utilisateur final, sont :
  - l'ordre de **tri physique de la table de faits** sur disque
  - et
  - la présence de résumés ou **d'agrégats préenregistrés**.



### Processus de conception FIN

- ✓ Le mini-ED doit respecter les exigences d'un processus métier déterminé.
- ✓ Assurer aussi une intégration aisée avec les autres mini-ED.
- ✓ constituer en définitive l'ED de toute l'entreprise.

Le tableau suivant illustre l'ensemble des mini ED de « **Maisons de Rêve** » .....

- Tables de faits et de dimension de chacun des processus métier de Maisons de Rêve.

Processus métier	Table de faits	Tables de dimensions
Ventes de propriétés	VentePropriété	Temps, Filiale, Personnel, PropriétéAVendre, Propriétaire, ClientAcheteur, Promotion
Location de propriétés	Bail	Temps, Filiale, Personnel, PropriétéALouer, Propriétaire, ClientLocataire, Promotion
Visites de propriétés	VisitePropriété	Temps, Filiale, PropriétéAVendre, PropriétéALouer, ClientAcheteur, ClientLocataire
Promotion de propriétés	Annonce	Temps, Filiale, PropriétéAVendre, PropriétéALouer, Promotion, Journal
Entretien de propriétés	EntretienPropriété	Temps, Filiale, Personnel, PropriétéALouer

## Propriété d 'additivité des faits

Additivité des Attributs de Fait :

Plusieurs millions de faits à résumer:

- compter les faits
- additionner les mesures

## Fait additif

Additionnable suivant toutes les dimensions

### Exemple

- quantités vendues,
- chiffre d 'affaire,
- coût,
- nombre de clients,
- nombre d'appel ...

### Fait semi-additif

Additionnable seulement suivant certaines dimensions

#### Exemple

- ✓ niveau de stock, de solde (valeurs instantanées)
  - excepté sur la dimension temps
- ✓ nombre de transaction, nombre de clients
  - excepté sur la dimension produit

### Fait semi-additif

- Soient deux faits (même magasin, même jour) (**Papier essuie tout, 20 clients**) et (**Mouchoir, 30 clients**)
- La somme du **nbr de clients** sur la DimProduit n'a pas de signification : un client peut avoir acheté des mouchoirs et du papier.
- sert uniquement de contrainte applicative
  - nbr de clients ayant acheté des mouchoirs (par mois)

## Fait non additif

Non additionable quelque soit la dimension

- comptage des faits

### Exemple

- un attribut ratio

$$\text{marge brute} = 1 - \text{Coût/CA}$$

## Dimension Temps

- Commune à tout entrepôt
- Relié à toute table de fait

Temps
time_key
day_of_week
daynum_in_month
daynum_overall
weeknum_in_year
weeknum_overall
month
quarter
year
holiday_flag
fiscal_period
event
season

## Dimension Temps

- L'importance des données temporelles est une des caractéristiques des SID qui les différencient des SI.
- Historiser (Archiver) les données :  
Les utilisateurs du DWH peuvent découvrir l'aspect de leur entreprise à n'importe quel moment ou période dans le temps.
- Découvrir et étudier des habitudes comportementales dans le temps et de faire des comparaisons entre des périodes similaires ou non-similaires.

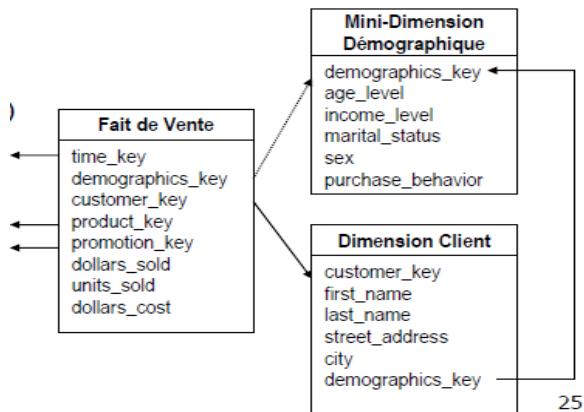
## Grandes Dimension

- Nombreux membres => réduire la taille des tables
  - dimension Produits (300.000)
  - dimension Clients (10.000.000)
  - Dimension humaine pour un pays : ~ 100'000'000
- **Solutions**
  1. L'appel du Flocon de Neige
    - tables de dimension secondaires (déportées) associée à une table de dimension
    - Faible gain de place et Navigation compromise
  2. Mini Dimensions
    - Mini dimensions démographiques pour les clients
- On peut éviter pas mal de problèmes des grandes dimensions et/ou des dimensions à évolution lente par l'ajout de mini dimensions.

## Grandes Dimension

### Mini Dimensions Démographiques : Dimension client

- nombreux enregistrements, nombreux attributs



## Dimensions Dégénérées

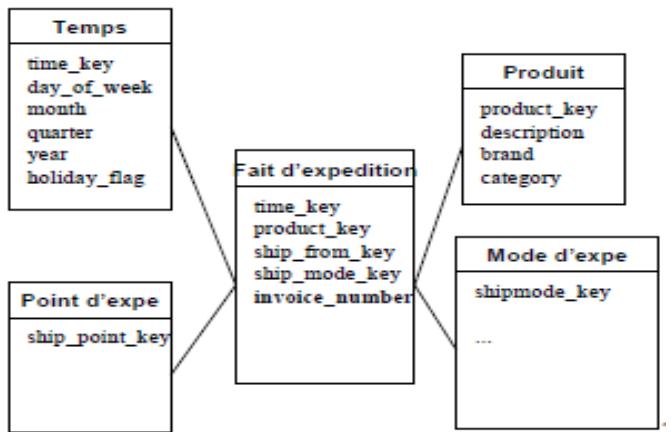
### • Dimension sans attribut

- Pas de table
- Mais la clé de dimension est dans la table de fait

### • Exemple

- numéro de facture (invoice number),
- numéro de ticket

## Dimensions Dégénérées



## Exemple 1: DWH d'un Supermarché

### • Dimensions

- **Temps** : 4 ans \* 365 jours = 1460 jours
- **Magasin** : 300
- **Produit** : 200000 références (10% vendus chaque jour)
- **Promotion** : un article est dans une seule condition de promotion par jour et par magasin

### • Fait

- $1460 * 300 * 20000 * 1 = 8,76 \text{ milliards d'enregistrements}$
- Nb de champs de clé = 4
- Nb de champs de fait = 4

Table des Faits =  $8,76 \cdot 10^9 * 8 \text{ champs} * 4 \text{ octets}$   
= **280 Go**

### Exemple 2: Suivi d 'appels téléphoniques

Temps : 3 ans \* 365 jours = 1095 jours

- Nb d 'appel par jour = 100 000 000
- Nb de champs de clé = 5
- Nb de champs de fait = 3

Table des Faits =  $1095 \cdot 10^8 \cdot 8 \text{ champs} \cdot 4 \text{ octets}$   
= **3,49 To**

### Exemple 3: Suivi d 'achats par carte de crédit

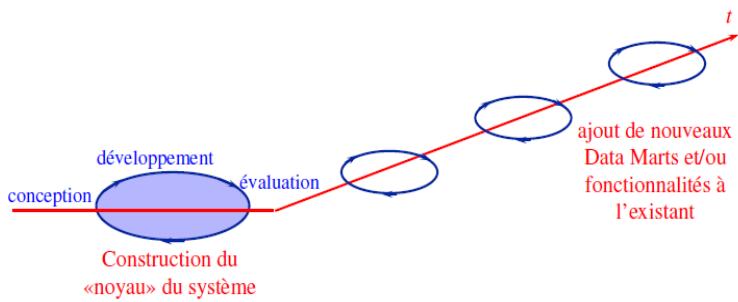
Temps : 3 ans \* 12 mois = 36 mois

- Nb de compte carte = 50 000 000
- Nb moyen d 'achat par mois par carte = 50
- Nb de champs de clé = 5
- Nb de champs de fait = 3

Table des Faits =  $54 \cdot 10^9 \cdot 8 \text{ champs} \cdot 4 \text{ octets}$   
= **1,73 To**

## Gestion de projet Data Warehouse

- Chaque Data Warehouse est **unique**
- Tâche **complexe et ardue**
- Construction **itérative**
  - Focalisations successives sur un ensemble de besoins



## Gestion de projet Data Warehouse

- **Le «sponsor»**
  - » membre de la direction, soutient le projet
- **Le comité utilisateur**
  - » différentes catégories (regroupement par besoins) des représentants
- **Les administrateurs du SI**
  - » très importants (connaissance des données)
  - » maintenance future du Data Warehouse
- **L'équipe de conception**
  - » souvent : consultants externes

## SQL Server 2008 BI Platform Components

Microsoft®  
SQL Server® 2008  
Integration Services

Microsoft®  
SQL Server® 2008  
Analysis Services

Microsoft®  
SQL Server® 2008  
Reporting Services

### Integrate

- Data acquisition from source systems and integration
- Data transformation and synthesis

### Analyze

- Data enrichment, with business logic, hierarchical views
- Data discovery via data mining

### Report

- Data presentation and distribution
- Data access for the masses

## Understanding BI Architecture

Source System



ETL System



Data Warehouse



Clients Tools



## Ateliers SQL Server 2008

Atelier 1 :

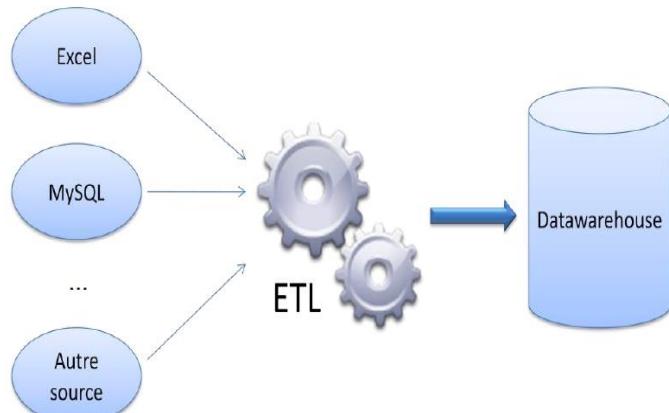
SSIS (Sql Server Integration Services)

Exemple de projets SSIS.

## Premières notions d'Intégration Services

- ETL (Extract Transform Load),
- Les packages,
- Les tâches d'intégration.

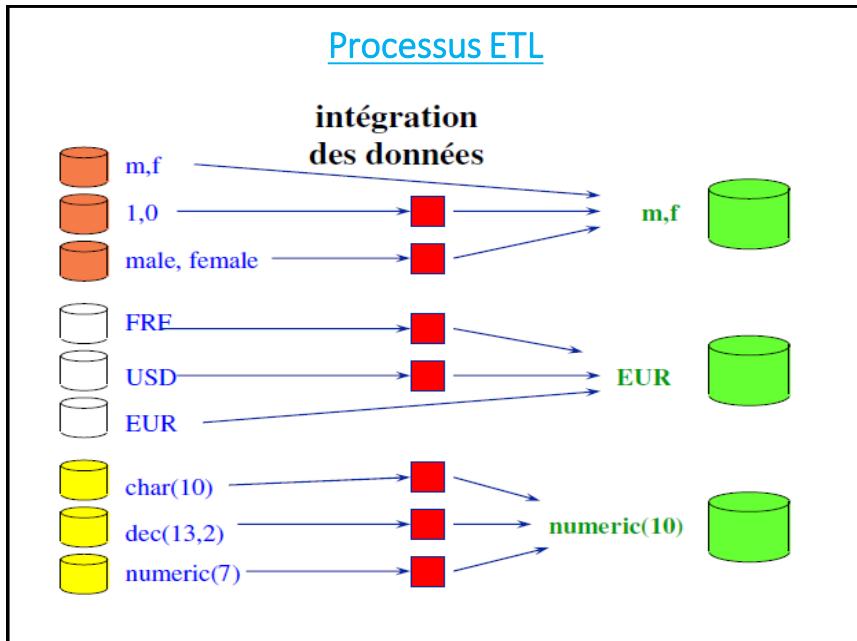
## ETL (Extract-Transform-Loading)



## Processus ETL

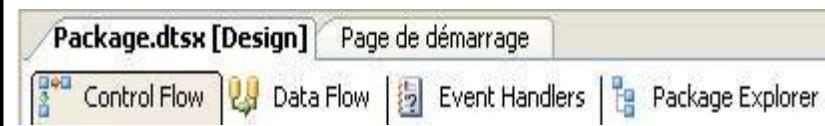
Ce processus se déroule donc en trois étapes :

- **Extraction** des données à partir d'une ou plusieurs sources de données telles que fichier plat, fichier brut, OLE DB (source relationnelles telles que SQL Server, Access...), Excel... ;
- **Transformation** des données agrégées ;
- **Chargement** des données dans la banque de données de destination (datawarehouse).



### Les composants d'un package SSIS

- **Le Control Flow ou flux de contrôle** : contrôler, d'ordonner et dissocier les tâches à réaliser par le package.
- **Le Data Flow ou flux de données** : la sélection, la transformation et l'insertion des données sont réalisées.
- **L'Event Handlers ou gestionnaire d'évènements** : gérer les évènements. EX: un traitement spécifique suite à une gestion d'erreurs.



## La boîte à outils SSIS

### Au niveau flux de contrôles:

Divers contrôles ou tâches sont pré-existants comme:

- les tâches de flux de données,
- nettoyage d'historique
- sauvegarde de BD, etc.



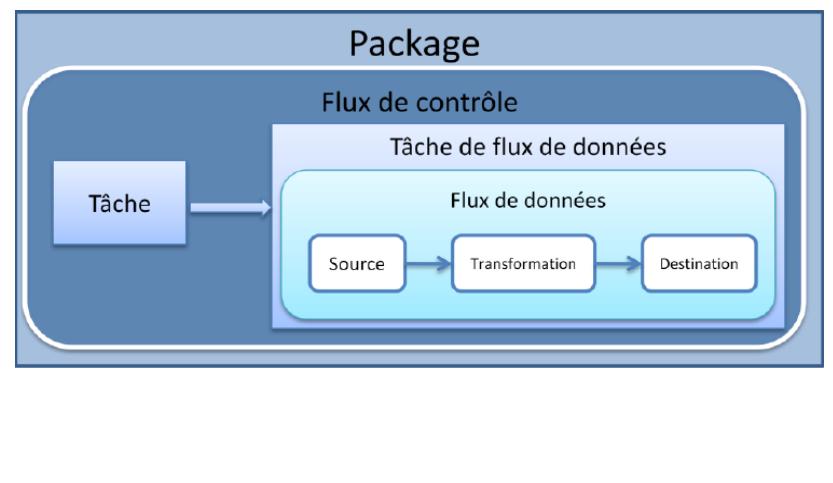
## La boîte à outils SSIS

### Au niveau flux de donnée:

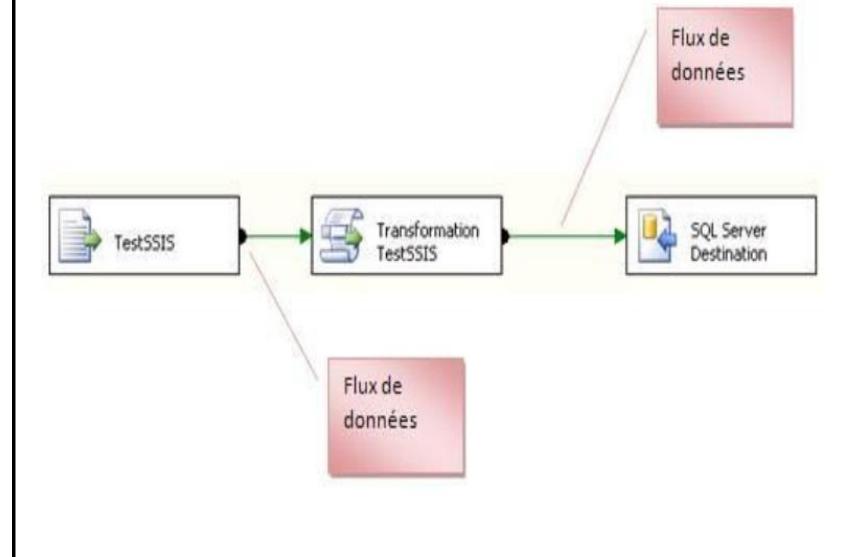
- **source** : les fichiers Excel, les fichiers plats, XML, etc.
- **transformation**:
- **destination**: fichiers plats, excel ou encore SQL Server,



## Les packages



## Exemple de projet SSIS



### Data Type : Date / Time

Data type	String format
DT_DBDATE	yyyy-mm-dd
DT_FILETIME	yyyy-mm-dd hh:mm:ss:fff
DT_DBTIME	hh:mm:ss
DT_DBTIME2	hh:mm:ss[.fffffff]
DT_DBTIMESTAMP	yyyy-mm-dd hh:mm:ss[.fff]
DT_DBTIMESTAMP2	yyyy-mm-dd hh:mm:ss[.fffffff]
DT_DBTIMESTAMPOFFSET	yyyy-mm-dd hh:mm:ss[.fffffff] [{+-} hh:mm]

## Ateliers SQL Server 2008

Atelier 2 :  
Quelques requêtes OLAP sur  
AdventureWorks.mdf

## OLAP : (On-Line Analytical Processing )

### Objectif:

#### ➔ Répondre à des Requêtes Analytiques

Rappel: Le DW intègre des données pour obtenir une vue d'ensemble des processus d'entreprise faisant l'objet d'analyses.

- On distingue ainsi entre :

- ✓ des requêtes **OLTP** (*Online Transaction Processing*) posées à des systèmes opérationnels et
- ✓ des requêtes **OLAP** (*Online Analytical Processing*) posées à des ED.

## Exemples de Requêtes OLAP

- **Navigation dans un cube**

“Nombre de livres d’enfants vendu en janvier, indépendamment du lieu”

- **Navigation à travers les niveaux de différente granularité**

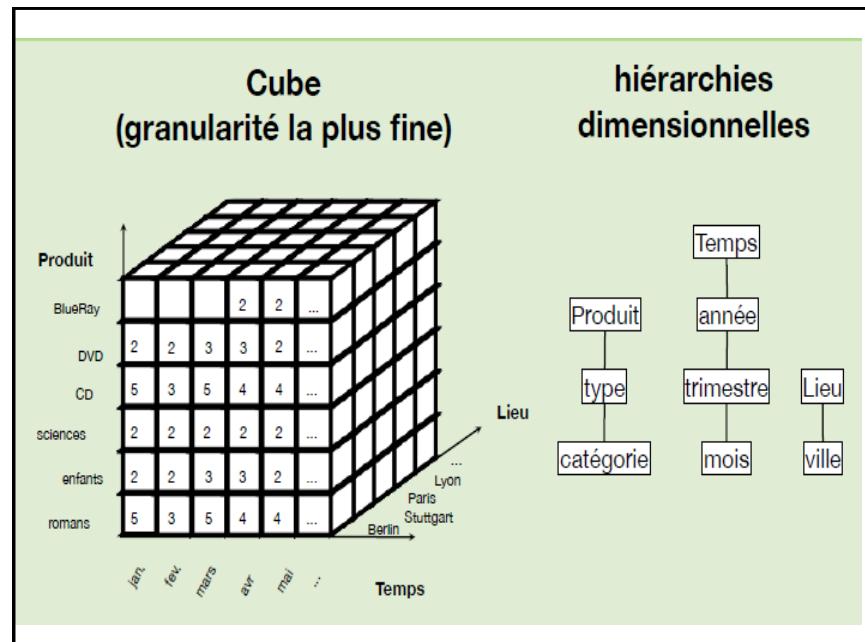
“CD par type de produit (musique, film, ...) par trimestre et ville”

- **Navigation à travers plusieurs cubes corrélos**

“Ventes totales en magasin et en ligne”, partant de deux cubes : (Produit/Temps/Lieu) et(Produit/Temps/Client).

- **Requêtes de ranking (classement)**

“Les 10 livres les plus populaires en 2009 par pays”.



## Vue d'Ensemble des Requêtes OLAP

Les requêtes abordées dans cette partie permettent la navigation dans le modèle multidimensionnel (MDDM – multi-dimensional data model).

- **Roll-up / Consolidate** : naviguer vers une granularité plus grossière.
- **Drill-down** : naviguer vers une granularité plus fine.
- **Drill-out / Split** : ajouter une ou plusieurs dimensions.
- **Drill-in / Merge** : réduire le cube d'une ou de plusieurs dimensions.
- **Slice** : sélectionner des données en appliquant un critère de sélection à une dimension.
- **Dice** : sélectionner des données en appliquant une sélection à plusieurs dimensions.

## Roll-Up

- Un roll-up correspond donc à un zoom-out du cube actuel, ce qui entraîne un résultat de granularité réduite.

### Ex 1 :

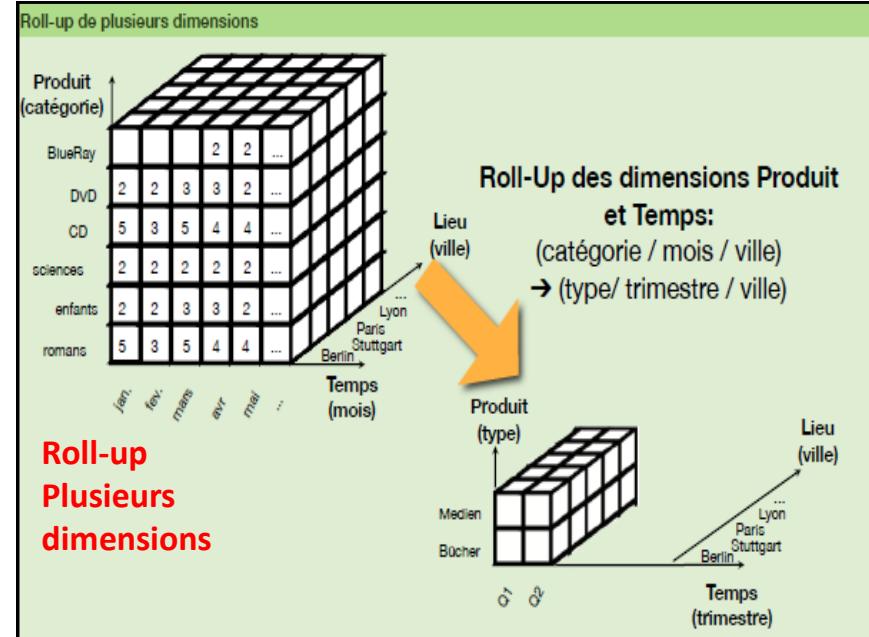
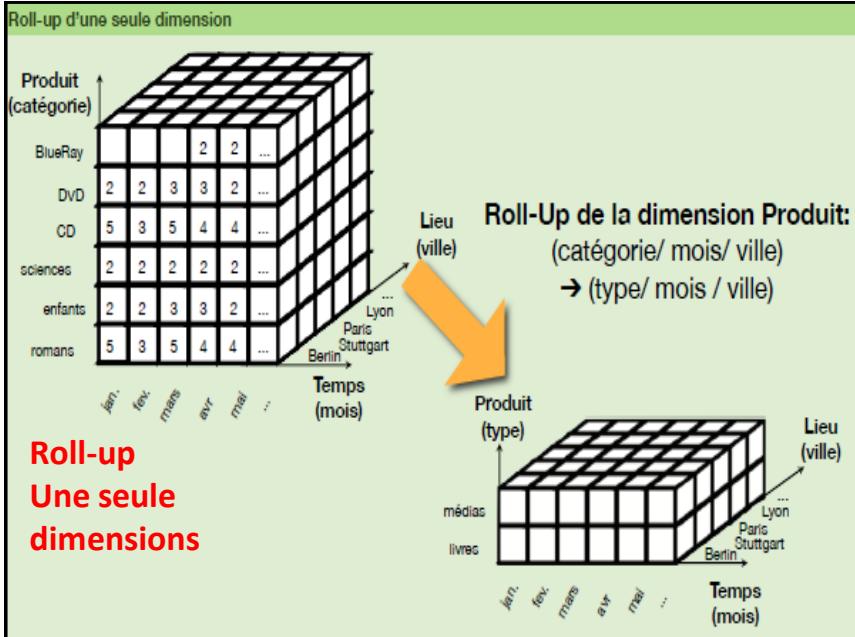
Roll-up du cube (**catégorie**/ mois/ ville) => (**type**/ mois/ ville).

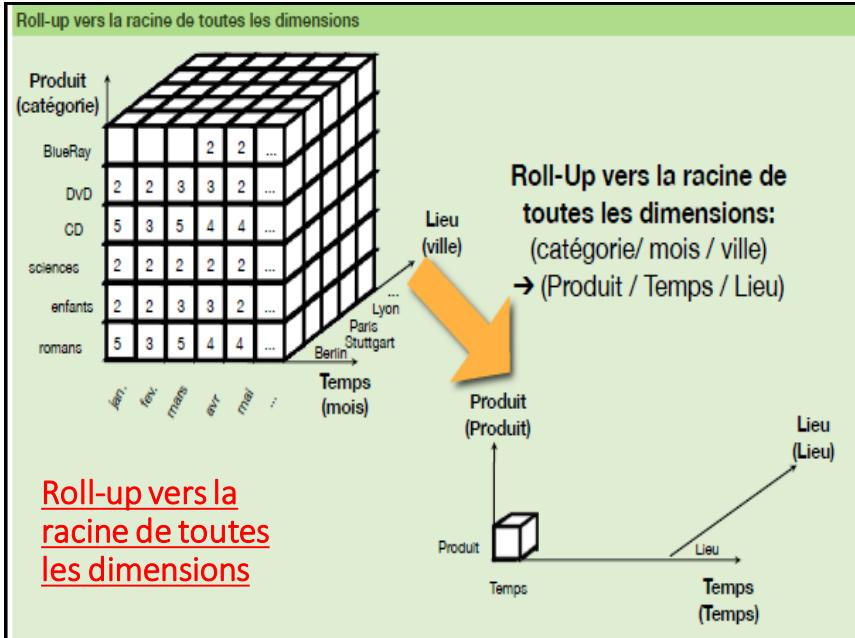
- Cette opération peut porter sur une ou plusieurs dimensions à la fois

### Ex 2 :

Roll-up du cube(**catégorie** / **mois** / ville) -> (**type**/ **trimestre**/ ville).

- Appliquée à toutes les dimensions, cette opération résulte en un cube n'ayant plus qu'un seul fait.





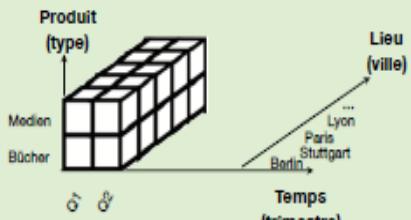
### Drill-down

- Drill-down est l'opération **inverse de Roll-Up**.
- navigation d'une granularité donnée vers une granularité plus fine pour une ou plusieurs dimensions (zoom-in).

### Ex:

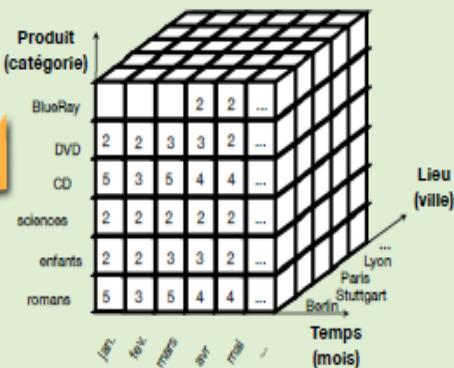
(*type / trimestre / ville*) => (*catégorie / mois / ville*).

### Drill-down de plusieurs dimensions



### Drill-down de plusieurs dimensions

**Drill-Down des dimensions**  
**Produit et Temps:**  
 (type/ trimestre / ville)  
 → (catégorie / mois / ville)



### Drill-in du cube 2D (catégorie / trimestre) vers le cube à une dimension (catégorie)

Produit (catégorie)	1	2	3	4
BlueRay	5	5	10	10
DVD	10	15	5	10
CD	3	3	4	4
sciences	5	4	3	3
enfants	5	5	5	5
romans	10	20	20	10

### Drill-In / Merge

### Drill-in de la dimension Temps

Janvier  
Février  
Mars  
Avril  
Mai  
Juin  
Juillet  
Août  
Septembre  
Octobre  
Novembre  
Décembre

Temps (trimestre)

Produit (catégorie)	1
BlueRay	30
DVD	40
CD	14
sciences	15
enfants	20
romans	60

## Slice : $\text{Slice}(p, D)$ :

- sélectionne les “tranches” le long de la dimension D, qui
- satisfont le critère de sélection p.

Le **paramètre** de l’opérateur **slice** est un **critère de sélection** (AND, OR et NOT)

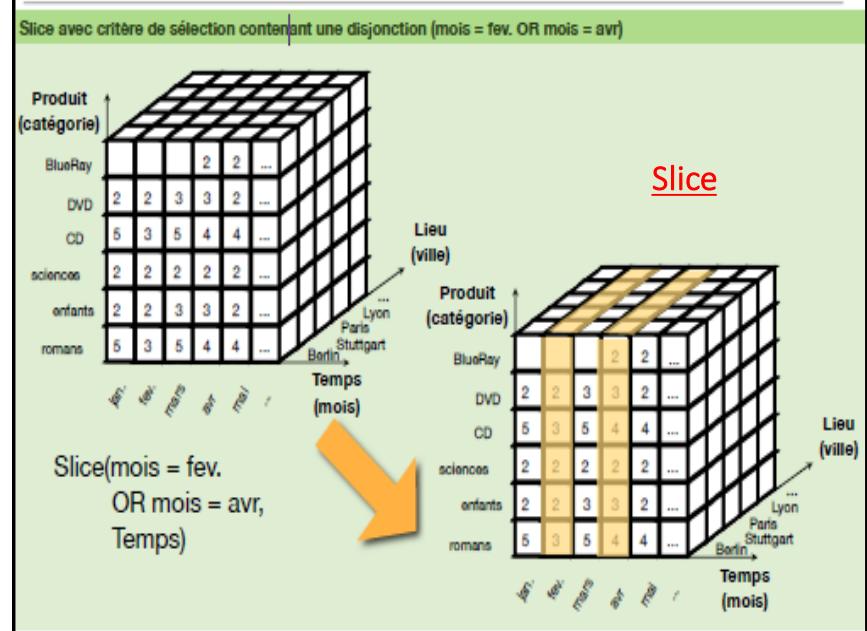
• Ex :

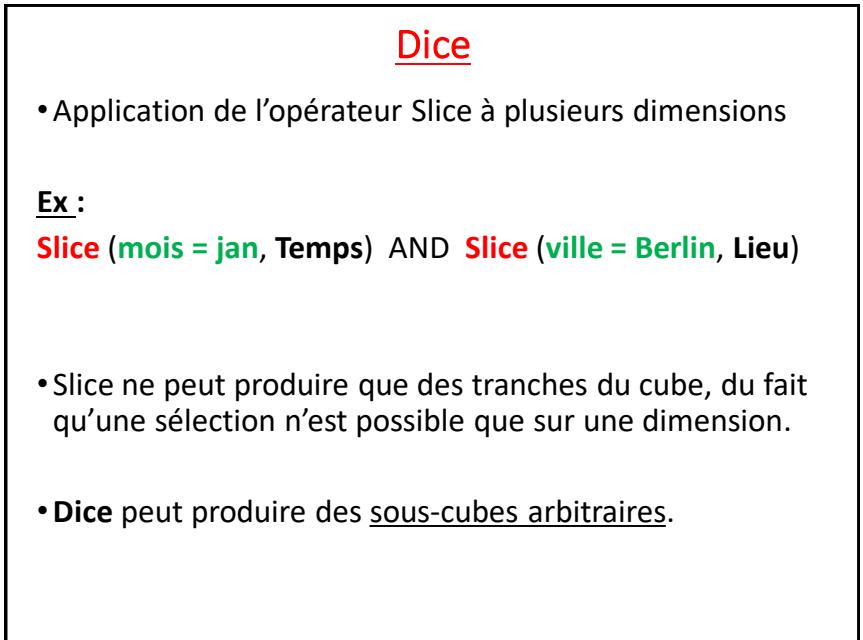
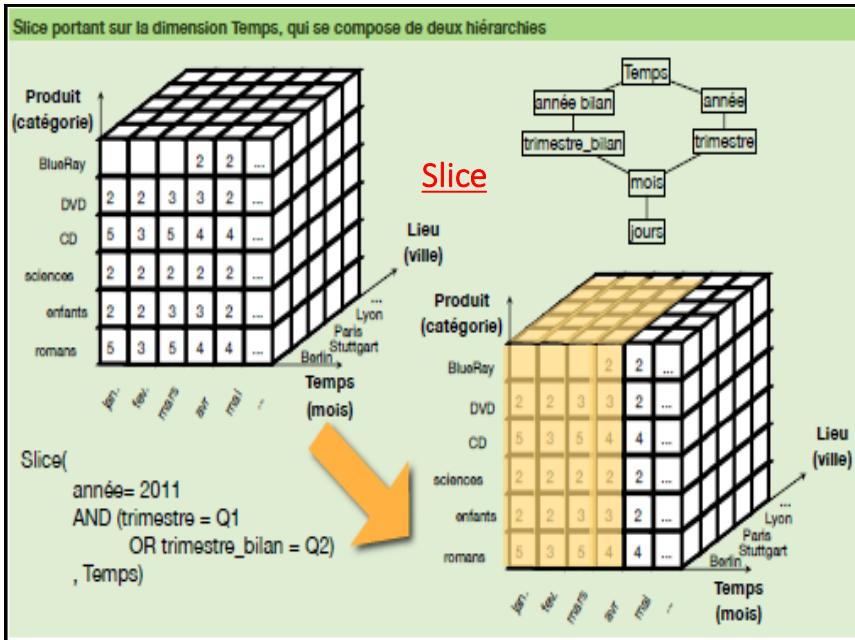
(mois > fev) AND (mois <= nov) AND (mois NOT avr)

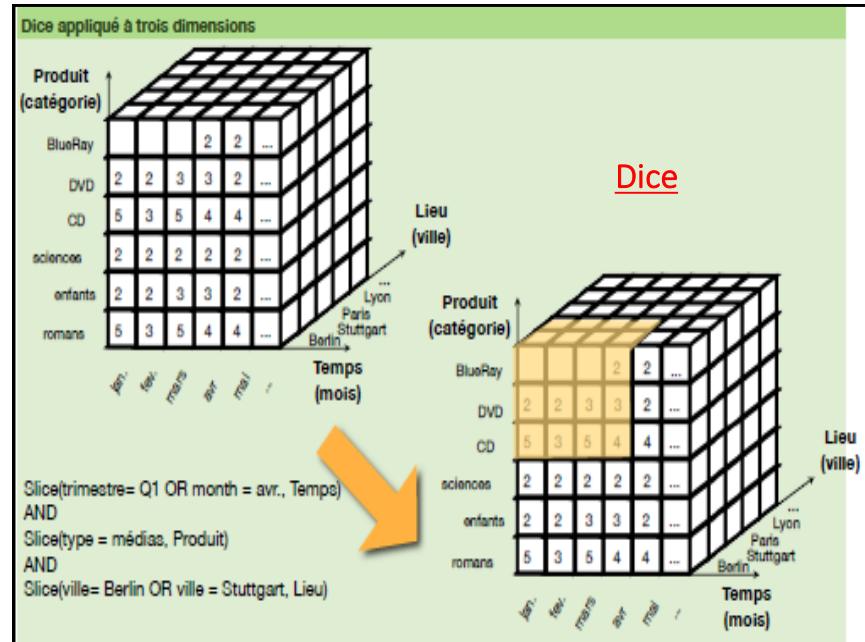
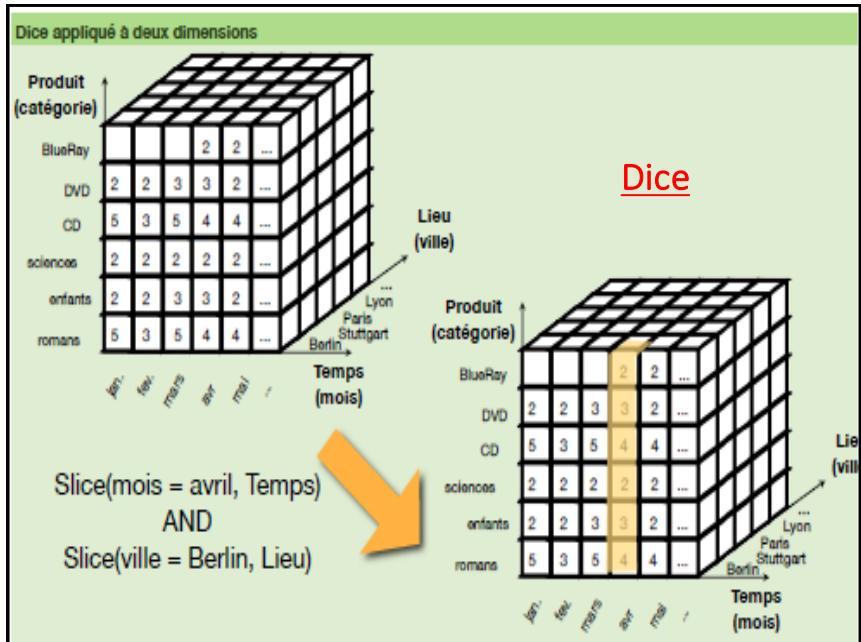
Critère de sélection sur un ou plusieurs niveaux **d'une même dimension.**

• Ex :

(pays = DE) AND (code\_postal = 10179)







## Ateliers SQL Server 2008

Atelier 3 :  
SSAS (Sql Server Analysis Services)

### SQL Server Analysis Services : SSAS

- la plateforme qui permet de créer et gérer :
  - ✓ des structures multidimensionnelles et
  - ✓ des modèles d'exploration de données.

Pour cela, il fournit :

- ✓ des fonctions **OLAP** (On Line Analytical Processing),
- ✓ et des applications d'exploration de données (**DataMining**).

Ces analyses comprennent un traitement sur des BD volumineuses et permettent de comprendre les métriques et les éléments qui influent sur le fonctionnement de l'entreprise.

## OLAP vs Data Mining

» OLAP : l'utilisateur cherche à confirmer des intuitions

Exemple:

«A-t-on vendu plus de yaourts en Région Parisienne qu'en Bretagne en 2003 ?»

» Data Mining : l'utilisateur cherche des corrélations non évidentes

Exemple:

«Quelles sont les caractéristiques de l'achat de yaourts ?»

## Adventure Works : Base de données exemple

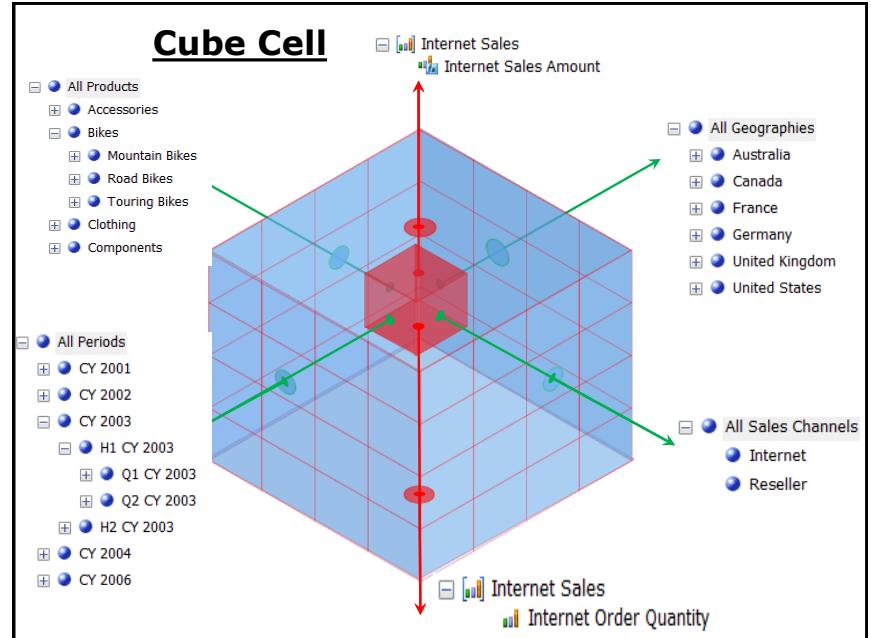
## La base de données exemple :

### Adventure Works Cycles,

la société fictive sur laquelle reposent les exemples de base de données **AdventureWorks.mdf**, est un grand fabricant d'envergure internationale.

L'entreprise :

- fabrique et vend des bicyclettes métalliques et des bicyclettes en alliage sur les marchés nord-américain, européen et asiatique.
- Le siège qui compte 290 employés est situé à Bothell dans l'état de Washington aux États-Unis et des équipes commerciales sont en place dans plusieurs des régions dans lesquelles la société a étendu son marché.

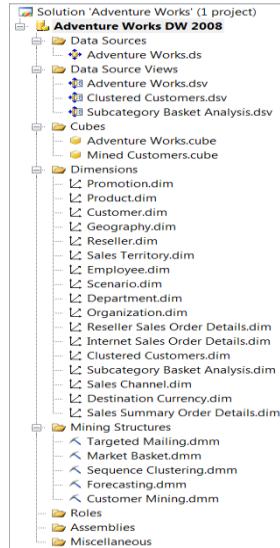


## Building Analysis Services OLAP Database

- Create Data Source
- ↓
- Create Data Source View
- ↓
- Create Dimensions
- ↓
- Create Measures**
- ↓
- Build Cube
- ↓
- Deploy Database to development server
- ↓
- Create Calculations and KPI

## Understanding Cube Representation

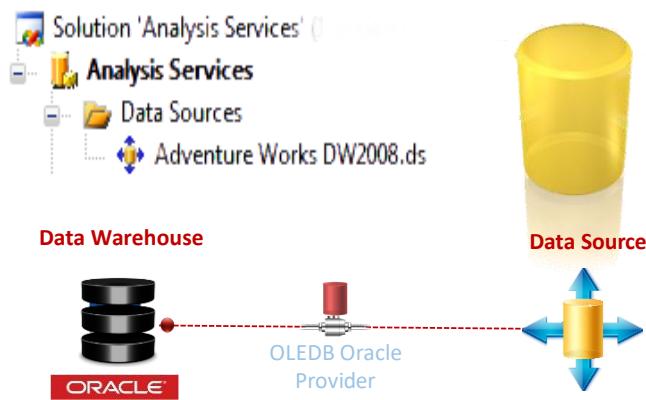
SSAS Project in BIDS



SSAS Cube  
Metadata

### Data Sources

Data sources est le point de départ pour votre activité de modélisation au sein de SSAS.

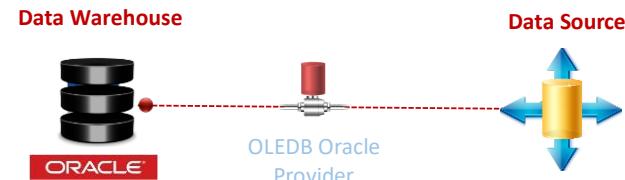


### Data Sources

OLE DB (parfois orthographié **OLEDB** ou **OLE-DB**) est une [API](#) développée par [Microsoft](#) permettant l'accès aux données.

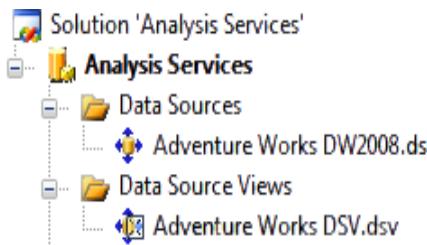
OLE DB se sert d'interfaces [COM](#) (Component Object Model).

Il a été conçu dans le but de remplacer [ODBC](#), de ce fait il permet l'accès à des BD non courante ou des sources de données qui n'utilisent pas un processeur de requêtes [SQL](#).

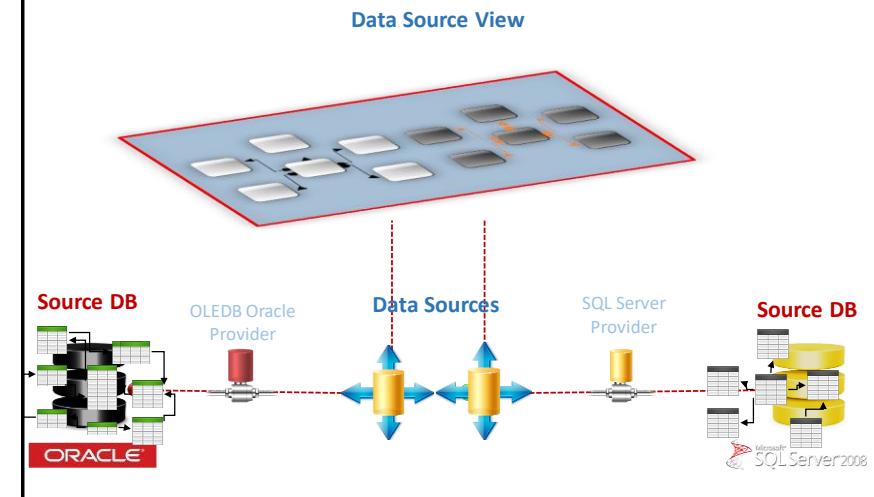


### Data source views

C'est une couche d'abstraction qui est utilisée pour étendre les objets (relational tables and views) qui sont exposées par la source de données à une collection d'objets à partir de laquelle des objets SSAS sont créés.



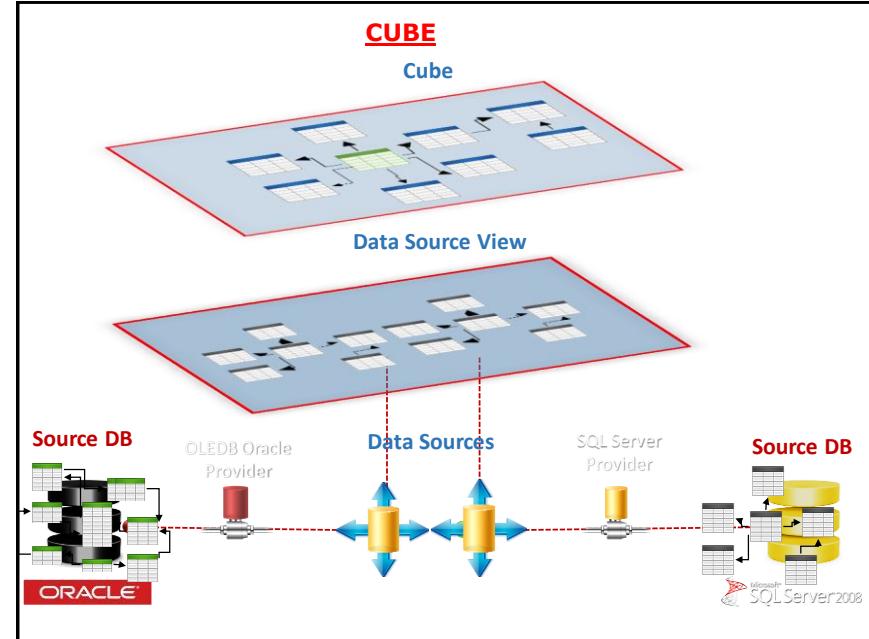
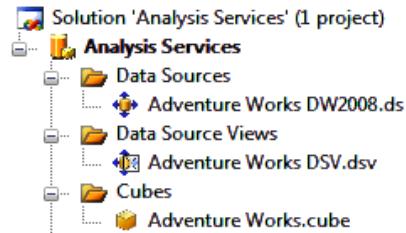
### Data source views



### Designing Cube

Un cube est une structure multidimensionnelle qui contient les dimensions et mesures.

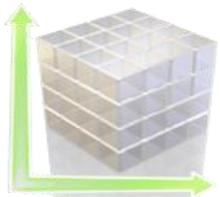
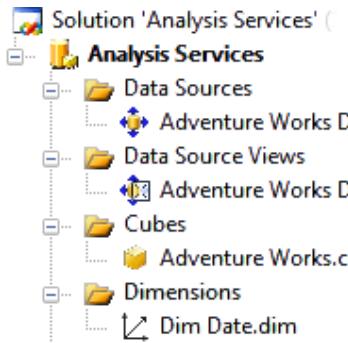
Les Dimensions définissent la structure du cube , tandis que les mesures fournissent les valeurs numériques d'intérêt pour l'utilisateur final.



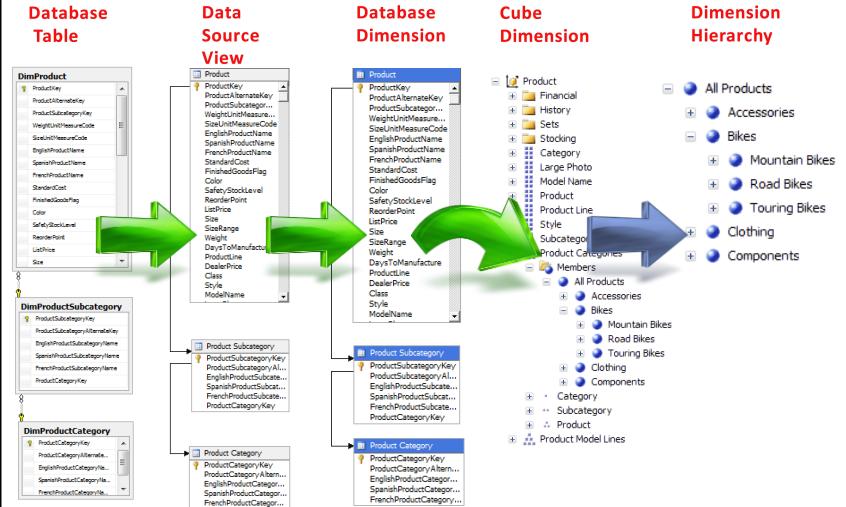
### Designing Dimensions

Les dimensions sont des attributs du cube. Ce sont des hiérarchies organisées et (niveaux) qui décrivent les données dans la table de faits

Ces catégories et niveaux décrivent des ensembles similaires de membres sur lesquels l'utilisateur veut fonder une analyse .



### Designing Dimensions



## Traitement et déploiement d'un cube

Après avoir fait des changements sur la structure du cube, on doit traiter le cube avant de tenter de parcourir ses données:

- Lorsqu'on traite un cube , les agrégations conçus pour le cube sont calculés et le cube est chargé avec les calculs et les données .
- Traitement d'un cube implique la lecture des tables de dimensions, la lecture de la table de faits, le calcul des agrégations spécifiées, et stockage les résultats dans le cube .
- Après qu'un cube soit traitée , les utilisateurs peuvent l'interroger.



## Ateliers SQL Server 2008

Atelier 4 :  
SSAS : Datamining  
les règles d'association

## Définition du datamining

- C'est le processus de découverte d'information à partir d'un large ensemble de données.
- Il utilise l'analyse mathématique pour extraire les modèles et les tendances qui existent dans les données.
- Ces infos ne peuvent pas être découvertes par une exploration traditionnelle des données,
  - ⇒ les relations sont trop complexes,
  - ⇒ il y a bq trop de données.

## Application datamining

- Le datamining peut être appliquer à des scénarios métier spécifiques comme:
  - **Prévision** des ventes
  - **Mailling ciblé** envers des clients spécifiques
  - Déterminer quels sont les produits susceptibles d'être **vendu ensemble**
  - Trouver les **séquences d'ordre** dans lesquels les clients ajoutent les produits à leurs paniers.

### Définition du problème

- l'analyse des besoins de l'entreprise
- la définition de la portée du problème,
- la définition des métriques par lesquels le modèle sera évaluée,
- la définition d'objectifs spécifiques pour le projet d'extraction de données.

### Préparation des données

- Le nettoyage des données
  - la suppression des mauvaises données,
  - de trouver des corrélations cachées dans les données,
  - l'identification des sources de données qui sont le plus précis,
  - déterminer les colonnes qui sont les plus appropriées pour une utilisation dans l'analyse.
- Des données incomplètes, des données erronées, et les entrées qui apparaissent séparés, et qui sont fortement corrélées, peuvent influencer les résultats du modèle de façon inattendu.

## Exploration des données

- Comprendre les données afin de prendre des décisions appropriées lorsque vous créez les modèles d'exploration.
- Les techniques d'exploration comprennent le calcul des valeurs minimales et maximales, à calculer les déviations moyennes et standard, et en regardant la distribution des données.
- Les écarts-types et autres valeurs de distribution peuvent fournir des informations utiles sur la stabilité et la précision des résultats. Un grand écart-type peut indiquer que l'ajout de données peut vous aider à améliorer le modèle.
- Les données qui dévie fortement avec une distribution standard pourrait être faussée, ou pourrait représenter une image précise d'un problème réel, mais il est difficile d'adapter un modèle aux données.

## Explorer et valider des modèles

1. Pour tester le fonctionnement d'un modèle
  - **Création de plusieurs** modèles avec différentes configurations
  - **faire des tests** sur ces modèles pour voir celui qui donne les meilleurs résultats
2. SSAS sépare les **données en apprentissage** et les **tests** afin d'évaluer avec précision la performance de tous les modèles sur les mêmes données
3. **Déployer** les modèles qui ont effectué le meilleur résultat dans un environnement de production

## Types d' algorithmes Datamining

### Algorithmes de Classification

- prévoir une ou plusieurs variables discrètes en se basant sur les autres attributs dans l'ensemble des données (dataset)

- Exemple

- Arbres de décisions (Decision Trees Algorithm)

## Types d' algorithmes Datamining

### Algorithmes de Régression

- prévoir une ou plusieurs variables continues, comme le **gain**, ou la **perte** en se basant sur les autres attributs dans le dataset.

- Exemple

Algorithme des séries temporelles (Time Series)

## Types d'algorithmes Datamining

### Algorithmes de Segmentation

- Diviser les données en groupes, ou clusters d'éléments qui ont des propriétés similaires.

- Exemple

- Clustering Algorithme

## Types of Data Mining Algorithms

### Algorithmes d'association

- Trouver les corrélations entre différents attributs dans le dataset.
- L'application la plus courante : les règles d'association qui sont utilisées dans l'analyse du panier de la ménagère.

- Exemple :

- Microsoft Association Algorithm

## Types of Data Mining Algorithms

### Algorithms d'analyse des séquence

- Résume des séquences fréquentes d'épisodes dans les données : flux de chemins web.

#### Exemple

- Sequence Clustering Algorithm

## Application des algorithmes

- Le choix du meilleur algorithme à utiliser pour une tâche métier spécifique est un vrai défi .
- Utiliser différents algorithms pour réaliser la même tâche métier, chaque algorithme produit un résultat différent.

Certains algorithmes peuvent produire plusieurs types de résultats

#### Exemple:

L'algorithme DT est utilisé pour la prévision, mais aussi pour réduire le nbre de colonne dans un dataset.  
car le DT peut identifier les colonnes qui n'affectent pas le model final.

Task	Algorithms to use
<b>Predicting a discrete attribute.</b> predict whether the recipient of a targeted mailing campaign will buy a product.	<ul style="list-style-type: none"> <li>Decision Trees</li> <li>Naive Bayes</li> <li>Clustering</li> <li>Neural Network</li> </ul>
<b>Predicting a continuous attribute.</b> forecast next year's sales.	<ul style="list-style-type: none"> <li>Decision Trees</li> <li>Time Series</li> </ul>
<b>Predicting a sequence.</b> perform a clickstream analysis of a company's Web site.	<ul style="list-style-type: none"> <li>Sequence Clustering</li> </ul>
<b>Finding groups of common items in transactions.</b> use market basket analysis to suggest additional products to a customer for purchase.	<ul style="list-style-type: none"> <li>Association</li> <li>Decision Trees</li> </ul>
<b>Finding groups of similar items.</b> segment demographic data into groups to better understand the relationships between attributes.	<ul style="list-style-type: none"> <li>Clustering</li> <li>Sequence Clustering</li> </ul>

## Les règles d'association

### Exemple d'application des règles d'association

- The Adventure Works Cycle company désire reconstruire les fonctionnalités de son site web.

Objectif :

Augmenter les ventes promotionnelles des produits.

comment :

1. identifier les ensembles de produits qui tendent à **être achetés ensemble.**
2. **prévoir les articles additionnelles** qui peuvent intéresser un client , en se basant sur les articles qui existent déjà dans son panier ou auxquels il a montré un intérêt (Analyse du panier de la ménagère)

### Les modèles d'association

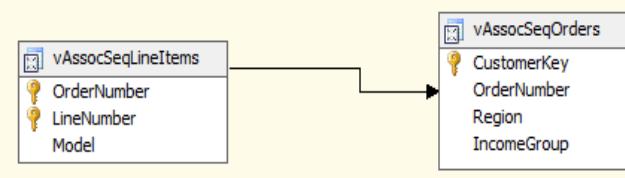
- sont construits dans des datasets qui contiennent **des clés et les articles (items)** contenus dans les **cas individuels** (une vente)
- un groupe d'Items dans un cas est appellé **itemset**
- Un modèle d'association =
  1. une série d'itemsets
  2. les règles qui décrivent comment ces itemset sont groupés ensemble dans un cas.

### Comment fonctionne l'algorithme

1. parcours un dataset pour trouver les articles(items) qui apparaissent ensemble dans un cas.
2. regroupe en itemset tout les articles associés, qui apparaissent au moins, dans le nombre des cas spécifiés par le paramètre **MINIMUM\_SUPPORT**.

### **Données requises pour les modèles d'association**

- **Colonne clé** : numeric ou texte qui identifie d'une façon unique chaque enregistrement. Les clés composites ne sont pas permises.
- **Colonne à prédire** : C'est la colonne clé de la table imbriqué, le champ produits achetés. Les valeurs doivent être discrète ou discretisé.
- **Colonne d'entrée** : doit être discrète. les données d'entrée sont souvent contenues dans 2 tables:



Une table contient les informations du client, tandis qu'une autre table contient les achats du client. on utilise alors une table imbriquée « **nested table** ».

### Paramètres du modèle

- **Support** : nbr de cas dans le dataset qui contiennent la combinaison des 2 items, X et Y.
- avec **MINIMUM\_SUPPORT** and **MAXIMUM\_SUPPORT**, l'algorithme contrôle le nbr d'itemsets générés.

- **Probability (confidence)**, la fraction (%) des cas dans le "dataset" qui contiennent X et contient aussi Y.

**MINIMUM\_PROBABILITY** permet de contrôler le nbr des règles générées

### Paramètres des règles générées

- L'**importance** mesure l'utilité d'une règle. Bien que la probabilité qu'une règle se produise puisse être élevée, son utilité peut être sans importance.
- => Plus l'importance est grande, plus la règle est importante.

### Exemple 1 :

#### Règle

"**if Touring1000 Tire= existing and Water bottle cage=existing, then Water bottle =existing**", avec probability=0.812

L'algorithme trouve que :

La présence dans le panier de "**Touring 1000 tire**" et "**water bottle cage**" prévoit que "**water bottle**" peut aussi être présente dans le panier.

### Exemple 2 :

#### un itemset :

"**Mountain 200 = Existing, Sport 100 = Existing**", avec support=710.

- L'algorithme génère ensuite les règles à partir des itemsets.
- Ces règles sont utilisées pour prédire la présence d'un item dans la BD, basée sur la présence d'autres items spécifiques que l'algorithme identifie comme importants.

## Exemple de règles générées

### Rule

Road Bottle Cage = Existing, Cycling Cap = Existing -> Water Bottle = Existing  
 Mountain-200 = Existing, Mountain Tire Tube = Existing -> HL Mountain Tire = Existing  
 Mountain-200 = Existing, Water Bottle = Existing -> Mountain Bottle Cage = Existing  
 Touring-1000 = Existing, Water Bottle = Existing -> Road Bottle Cage = Existing  
 Road-750 = Existing, Water Bottle = Existing -> Road Bottle Cage = Existing  
 Touring Tire = Existing, Sport-100 = Existing -> Touring Tire Tube = Existing

## Résultat de l'atelier 4 : règles d'associations

Mining Model:	v Assoc Product 1	Viewer:	Microsoft Association Rules Viewer
Rules	Itemssets	Dependency Network	
Minimum probability:	0.40	Filter Rule:	
Minimum importance:	0.59	Show:	Show attribute name only
<input type="checkbox"/> Show long name		Maximum rows:	20
Probability	Importance	Rule	
1.000	1.310	Mountain-200 Black, 42, Mountain Tire Tube -> HL Mountain Tire	
1.000	1.234	Road-750 Black, 58, Water Bottle - 30 oz. -> Road Bottle Cage	
1.000	1.205	Fender Set - Mountain, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.234	Road-750 Black, 44, Water Bottle - 30 oz. -> Road Bottle Cage	
1.000	0.822	Road Bottle Cage, Sport-100 Helmet, Red -> Water Bottle - 30 oz.	
1.000	1.232	Road-750 Black, 48, Water Bottle - 30 oz. -> Road Bottle Cage	
1.000	0.832	Road Bottle Cage, AWC Logo Cap -> Water Bottle - 30 oz.	
1.000	1.238	Road-750 Black, 52, Water Bottle - 30 oz. -> Road Bottle Cage	
1.000	0.822	Road Bottle Cage, Sport-100 Helmet, Blue -> Water Bottle - 30 oz.	
1.000	1.149	Mountain-200 Silver, 42, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.150	Road Bottle Cage, Sport-100 Helmet, Black -> Water Bottle - 30 oz.	
1.000	1.289	Mountain-200 Silver, 46, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.152	Touring Tire, Sport-100 Helmet, Red -> Touring Tire Tube	
1.000	1.284	Mountain-200 Black, 38, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.152	Hydration Pack - 70 oz., Touring Tire -> Touring Tire Tube	
1.000	0.818	Mountain-200 Silver, 38, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.153	Hydration Pack - 70 oz., Road Bottle Cage -> Water Bottle - 30 oz.	
1.000	0.773	Mountain Tire Tube, Water Bottle - 30 oz. -> Mountain Bottle Cage	
1.000	1.151	LL Road Tire, Sport-100 Helmet, Red -> Road Tire Tube	
		Mountain-200 Black, 42, Water Bottle - 30 oz. -> Mountain Bottle Cage	

## Les arbres de décisions

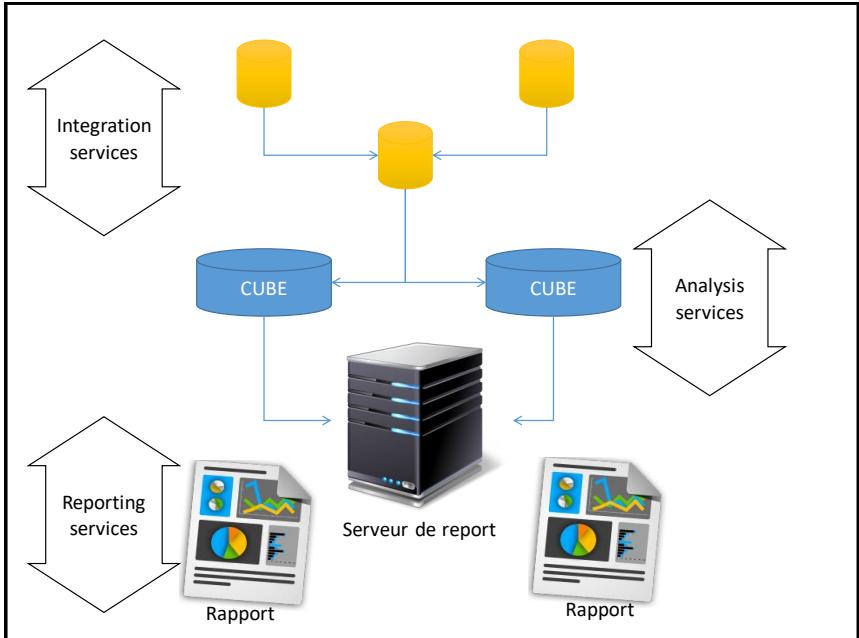
## Ateliers SQL Server 2008

Atelier 5 :  
SSAS : les arbres de décision

## Atelier 6: SSRS- Sql Server Reporting Services

### Reporting

- Le principe du *reporting* est d'agréger et de synthétiser des données nombreuses et complexes sous forme d'indicateurs, de tableaux, de graphiques permettant d'en avoir une appréhension globale et simplifiée.
- Le *reporting* s'appuie principalement sur les agrégats (GROUP BY en SQL par exemple) afin de faire apparaître des comptages, sommes ou moyennes en fonction de critères d'analyses.
- Le *reporting* est généralement récurrent, le même rapport sera produit à intervalles réguliers pour contrôler les variations des indicateurs.



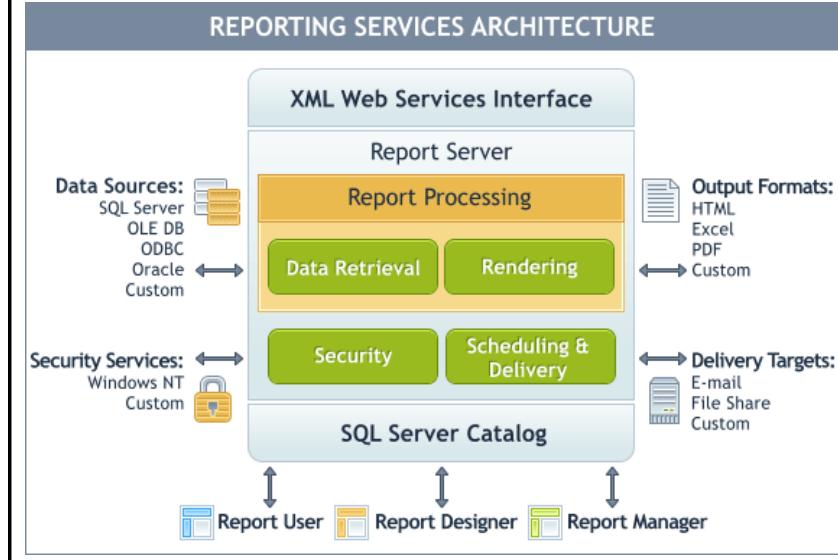
### Concepts de RS

- SSRS fournit une gamme complète de services et d'outils prêts à l'emploi pour créer, déployer et gérer des rapports.
- SSRS intègre des fonctionnalités de programmation qui permettent d'étendre et de personnaliser la création de rapports.

## Concepts de RS

- Les rapports sont développés via Visual Studio pour être publiés sur un portail Web.
- **SSRS** inclut aussi un outil de génération de rapports **Report Builder**, un concepteur de rapports intuitifs à destination des utilisateurs finaux.

## Architecture



## Fonctionnalités de RS

- SSRS est une plateforme serveur permettant :
  - La centralisation de la gestion et du stockage des rapports.
  - la gestion de la sécurité d'accès aux rapports en se basant sur des rôles.
- Création de différents types de rapports
  - ✓ interactifs,
  - ✓ tabulaires,
  - ✓ graphiques,
  - ✓ matrices,
  - ✓ tableaux de bord,
- Création de rapports à partir de différentes sources de données (Excel, Fichiers plats, Oracle, Ole DB, ODBC, XML...).

## Fonctionnalités de RS

- Exportation des rapports vers d'autres applications.
- Publication des rapports via le portail web SSRS ou portail SharePoint ou par le biais d'autres applications spécifiques
- Navigation optimisée à travers une bonne structuration des rapports volumineux.
- Création des rapports ad-hoc basés sur des modèles prédéfinis via Report Builder.

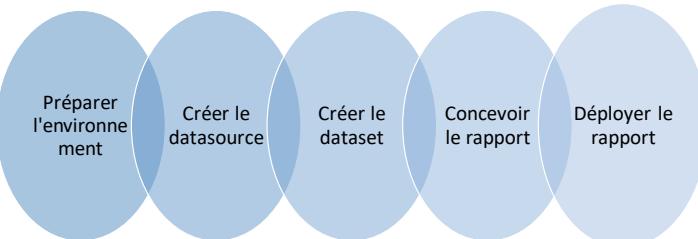
## Avantages

- Un faible coût d'acquisition et de mise en place par rapport aux autres outils de reporting disponibles sur le marché (parfait pour les PME).
- Mise en place simple et rapide d'infrastructures de reporting centralisées .
- Standardisation de l'utilisation des rapports
- Le développement SSRS ne nécessite pas des compétences techniques pointues
- Les objets SSRS développés (rapports, sources de données et datasets) sont sous format XML.

## Avantages

- Le développement SSRS se fait via Visual Studio donc la création d'applications spécifiques utilisant des rapports se fait dans un même environnement (Exemple : Générer des rapports à partir d'une application Net personnalisée).
- Possibilité d'exporter les rapports dans plusieurs formats (Excel, Word, Texte, XML, PDF, TIFF, MHTML, ...)
- Intégration dans SharePoint.

## Déploiements des rapports



## Déploiements des rapports

- Préparer l'environnement
  - Installation de SQL Server 2012
  - Installation de Visual Studio
  - Configuration du serveur
  - Création de la base de données
  - Importation des données

## Déploiements des rapports

- Créer le datasource
  - Déterminer le serveur de base de données
  - Choisir la base de données

## Déploiements des rapports

- Créer le dataset
  - Déterminer le datasource
  - Spécifier la requête
    - Text
    - Query Designer
      - Tables
      - Champs
      - Jointures
      - Réstrictions

## **Déploiements des rapports**

- Concevoir le rapport
  - Choisir le type de rapport(graphique, tableau, liste, carte,...)
  - Implémentation des données depuis le dataset

## **Démonstration**

- Découverte de l'outil
- Résultat attendu
- Créer un rapport
- Déterminer la source de données
- Spécifier les requêtes (dataset)
- Conception et design du rapport
- Test

## **Ateliers SQL Server 2008**

Atelier 6 :  
SSRS (Sql Server Report Services)

## **Projets SQL Server 2008**

### Projet 1 : Création du datamart InternetSales sous SQL Server 2008

- Importer la base de données **Sales** dans SSMS 2008.
- Créer le schéma de la base de données
- Concevoir l'entrepôt de données correspondant
- Utiliser **SSIS** pour intégrer les données sources vers le datawarehouse, sous SQL Server 2008
- Vérifier et tester les données avec des requêtes **MDX**.

### Projet 2 : Création du datamart Finance sous SQL Server 2008

- Importer la base de données **Finance** dans SSMS 2008.
- Créer le schéma de la base de données
- Concevoir l'entrepôt de données correspondant
- Utiliser **SSIS** pour intégrer les données sources vers le datawarehouse, sous SQL Server 2008
- Vérifier et tester les données avec des requêtes OLAP (**MDX**).

### Projet 3 : Ajouter des objets au cube d'Analyse de l'Atelier 3

- Ajouter des Calculs
- Ajouter des KPI
- Ajouter des Actions
- Construire des rapports sous forme de tableaux de bords.

### Projet 4 : Création du datamart ResellerSales sous SQL Server 2008

- Importer la base de données **ResellerSales** dans SSMS 2008.
- Créer le schéma de la base de données
- Concevoir l'entrepôt de données correspondant
- Utiliser **SSIS** pour intégrer les données sources vers le datawarehouse, sous SQL Server 2008
- Vérifier et tester les données avec des requêtes OLAP (**MDX**).