Modélisation des ED : Concepts de base

Modélisation des BD : Entité et relation.
Modélisation des DW : dimension et mesure.
-**les dimensions** : les points de vues depuis lesquels les mesures peuvent etre observées/ une table qui contient les axes d'analyse selon lesquels on veut étudier des données observables.
-**les mesures** : valeurs numériques que l'on compare. C'est le resultat d'une operation d'agrégation des données
-**les propriétes des mésures** :
- Fait additif : additionable suivant toutes les dimensions. ex qantité vendue - chiffre d'affaire.
- Fait semi-additif : additionable selon certain les dimensions. ex niveau de stock - nombre de transactions.
- Fait non-additif : non-additionable, recalculer. ex MxCa pour l'ensemble des magasins.

-Les faits :

-**Un fait** represente un sujet d'analyse, la valeur d'une mesure, mesurée ou calculée, selon un membre de chacune des dimensions.
-**Les mesures** sont stockées dans les tables de faits.
-**La table de fait** contient les données observables(les faits)/les mesures, les clés vers les tables de dimensions, les dimensions dégénérées(sans attribut).
-Dans un **entrepot de données** les "Faits" sont normalement numériques puisqu'elle sont d'ordre quantitatif.

-**Caractéristiques d'une table de faits** :
-contient les valeurs numérique de ce qu'on désir.
-contient les clés étrangères.s
-contient un nombre de colonnes réduit.
-contient plus d'enregistrement qu'une table de dimensions
-Caractéristiques des info de la table de faits :
-elles sont numériques et sont utilisées pour faire des SUM, AVG
-les données sont additives/semi-additives
-les mesures ds la table doivent référer et avoir un lien direct aux clés de dimension.
-**une dimension peut être définie comme** un thème ou un axe selon lequel les données serons analysées. C'est un axe d'analyse
-**une dimension contient** des membres organisées en hiérarchie. (temps: année, semestre, mois, jour).
-**structure de base d'une table de dimension** :
=>Clés de substitution(cléprimaire)/Clés d'affaires(clénatuturelle), attributs de la dim , Clés spéciales (pour la gestion de l'historique de la dimension).

-**Caractéristiques d'une dimension**:
-contient le detail sur les faits.
-contient les info descriptives des valeurs numériques de la table de faits.
-les attributs sont souvent utilisés comme "lignes" et "colonne" dans un rapport ou résultat de requéte.

=>**Clé de substitution** : clé non-intelligente utilisée dans un DW pour remplacer et compléter la clé artificielle du sys opérationnel afin de rendre un elemnet unique dans la dimension.
=>**Clé naturelle** est en generale composée de plusieurs colonnes.
-**Dans un systeme opérationnel on utilise une clé artificielle** afin d'identifier d'une facon unique un element de l'entité.
=>La clé de substitution ne doit pas etre confondue avec la clé artificielle attribuée par le sys opérationnel.

Fonctionnalité des clés de substitution :
- Remplacer la clé naturelle : la cle de substitution remplace la clé artificielle
- Completer l'information : la clé de substit est utilisée dans l'entrepot de données seulement

Avantages des clés de substitution :
-Performance : accélère l'accès aux données du moment ou l'on va utiliser un index numèrique
-Indépendance du sys source : on ne peut garantir que la clé d'affaire ne change pas dans les systèmes sources

-Historique des changement et granularité infinie : gérer la cle de substitution pour garder l'historique des changements de la dimension selon certains critères

---

Table d'une BD multidimensionnelle :

Les dimensions: Clés spéciales:
-Date effective: date du creation de l'enregistrement.
-Date retrait : date ou l'enregistrement a été retiré.
-Indicateur effectif: "O" si l'enregistrement actif, "N" sinon.
(facultatif) :
-Type de l'evolution (SCD)
-Valeur avant le changement

Evolution des dimensions :

-Lente : gestion de la solution
-> Ecrasement de l'ancienne valeur: (correction des info erronées)
-facile a mettre en œuvre (avantage)
-perte de la trace les val antérieures des attributs. (inconvenient)
-perte de la cause de l'evolution ds les faits mesurés. (inconvenient)
-> Versionnement (ajout d'un nouvel enregistrement, utilisation clé primaire)
-permet de suivre l'evolution des attributs (avantage)
-permet de segmenter la table de faits en fonction de l'historique. (avantage)
-accroit le volume de la table. (inconvenient)
-> Valeur d'origine/courante (ajout d'un nouvel attribut) :
-avoir deux visions simultanées des données. (avantage)
-voir les données comme si le changement n'avait pas eu lieu (avantage)
-inadapté pour suivre plsr valeurs d'attributs intermédiaires. (inconvenient)

-Rapide :
-> Subit des changement très fréquents dont on veut préserver l'historique
-> Solution : isoler les attributs qui changent rapidement.

Modélisation des ED : Modèle d'un DW

-Modèle en étoile : table de faits au centre du schéma
-Modèle en flocon de neige
-Modèle en constellation/galaxy

Modèle en étoile :
-> Avantages : - Facilité de navigation
- Performances (nb de jointure limité)
-> Inconv : - Redondances dans les dimensions
-Alimentation complexe

Modèle en flocon de neige :
=>les dimensions sont décomposées en hiérarchie , c'est une methode de normalisation des tables de dimensions.
=> le seul but est de minimiser l'espace disque sinno il n'est pas recommandé.
=> si les tables sont très volumineuses-> reduction du volume + possibilité de réaliser des analyses pa pallier (drill down).
-> Avantages : - Concu pour des requetes flexibles sur des dimensions et des relations complexes
- Performence des requetes
- Grande evolutivité
-> Inconv : -requetes plus complexes.
-plus de temps d'excution.
-navigation difficile.
-nombreuses jointures.
-maintenance supplémentaires.

Modèle en constellation/galaxy:
=>Fusionner plsr modèles en étoile qui utilisent des dimensions communes.
=>Structure logique d'entrepot de données.

Définition de TALEND Talend est un ETL (Extract Transform and Load) qui permet d'extraire des données d'une source, de modifier ces données, puis de les recharger vers un destination. La source et la destination des donnéees peuvent être une base de données, un service web, un fichier csv. et bien d'autres…Talend peut donc être utilisé dans n'importe quel contexte ou des données sont véhiculées.

Démarche de conception :

étape 1 : -choisir le processus à modéliser. Processus Métier
étape 2 : -choisir le grain de faits, données granulaire.
-décider de ce que represente une ligne de la table de faits Fait à observer
-niveau de detail: transaction individuelles,..
étape 3 : -identifier les dimensions. (typiquement le temps, le client...). Dimensions
étape 4 : -identifier les mesures de faits (de préférence additives). Mesures

---

Analyse Multidimensionnelles

Le Data Warehouse est le socle indispensable pour obtenir les réponses aux questions essentielles à la prise de décision et au pilotage de l'entreprise.
Quitter un modèle de base de données opérationnelle et se baser sur un modèle de base de données décisionnelle revient à : transformer les éléments du modèle de base de données opérationnelle en dimensions et fait

BD Multidimensionnelles : Cube

-Un cube OLAP est une structure de données multidimensionnelle stockant les faits comme des mesures indexées par plusieurs dimensions.
-Chaque cellule d'un cube représente la mesure ou valeur quantitative d'un fait sur le croisement de plusieurs dimensions.
-L'intérêt d'un cube OLAP est d'offrir à l'utilisateur la capacité de faire des analyses multidimensionnelles ou des agrégations par axe de dimension dans l'espace.

BD Multidimensionnelles = super-tableur.

BD multidimensionnelle, Opérateurs :

-Les opérateurs appliqués sur un cube sont algébriques.
-Le résultat est un autre cube.
-Les opérateurs permettent :
-Des extractions Slicing : Prendre une tranche du cube.
-Des extractions Dicing : Extraire un sous-cube.
-Des changements de granularité d'une dimension :
– Roll-up (agrégation d'une dimension -> Résumé)
– Drill-down (informations plus détaillées).

Technologies OLAP

• **ROLAP** - Relationnel OLAP : OLAP sur du relationnel
• **MOLAP** - Multidimensionnel OLAP : OLAP sur un DW dimensionnel
• **HOLAP** - Hybride OLAP : Mélange des deux

=>ROLAP :
-Utilisent un SGBD relationnel classique avec des adaptations spécifiques à l'OLAP
-organisées en schémas en forme d'étoiles ou en flocon de neige.
-Peuvent conduire à des temps de réponses élevés.
-s'appuie sur la maturité de la tech relationnelle

-N'effectuent pas des requêtes OLAP avec des calculs complexes.
-Il réduit le nombre de jointures à réaliser pour l'exécution d'une requête.
-lent et peu performant mais sans limites de taille.
-Exemples : IBM DB2 , Oracle, SQL Server, Informix, ...

=>MOLAP (OLAP multidimensionnel)
-Utilisent un SGBD multidimensionnel.
-Les données sont stockées directement ds une structure de Cube multidimensionnel dont : les structures sont optimisées + l' accès est rapide en lecture/ écriture.
-MOLAP necessite le pre-calcul, sur tous les niveaux de hiérarchies des dimensions.
-Très rapide et performant mais avec des limitations de taille
-Exemples : Essbase, SAS OLAP Server, …

=>HOLAP (HOLAP permet d'avoir des DW de taille importante tout en ayant des temps de réponse satisfaisants)
-Les données multidimensionnelles sont stockées et traitées en se basant sur le SGBD Relationnel et le SGBD multidimensionnel, afin d'éviter les problèmes des systèmes MOLAP et ROLAP.

-L'approche HOLAP consiste à
• Utiliser les tables comme structure permanente de stockage des données
• Manipuler les informations du DW avec un moteur ROLAP
• Exploiter les Data Marts selon une approche multidimensionnelle avec un système MOLAP.

-Combinent les points positifs des systèmes ROLAP et MOLAP :
• stocker les données détaillées du DW dans un SGBD Relationnel
• stocker les données agrégées, souvent des magasins de données (Data Marts) du DW dans un SGBD Multidimensionnel

-HOLAP permet d'avoir des DW de taille importante tout en ayant des temps de réponse satisfaisants.
-Exemples : Oracle, IBM DB2 OLAP Server …

---

générer des requetes plus compacte que les requetes SQL
-> select axis1 ON COLUMNS, axis2 ON ROWS, axis3 ON AXIS(0).

from
where (slicer)
Membre = une instance d'un niveau d'une dimension.
Tuple = suite de membres entre () séparés par une virgule.

Cellule = le tuple permet d'identifier les cellules ds un cube.
Mesure = ds un tuple les mesures sont traitées comme une dimension particuliére.
Set = ensemble ordonné de tuples définit sur une meme dimension.(le mot apres ON)

---

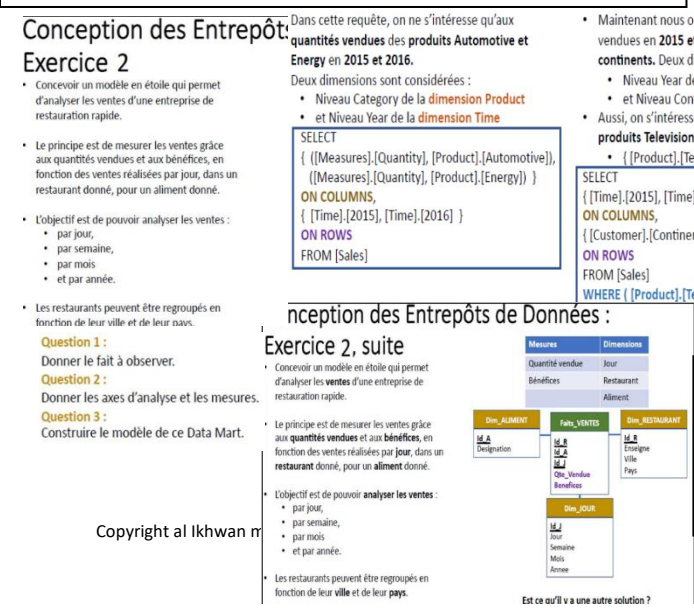
pour définir un axe et présenter sur l'axe tous les membres d'une dimension :
=> <dimension name>.MEMBERS
pour voir apparaitre tous les membres d'une dimension a un certain niveau :
=> <dimension name><level name>.MEMBERS
Ds un slice on peut avoir plsr membres, mais ils doivent appartenir a des dimension différentes.
// ou -- => commentaire en fin de ligne.
/* */ => commentaire sur plsr lignes

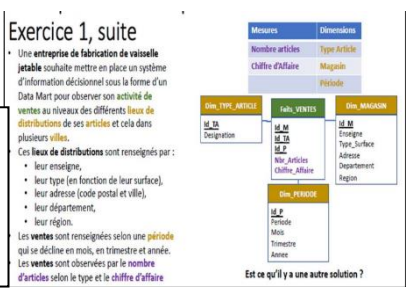
SQL	MDX
SELECT column1, column2, ..., columnn **FROM** table	**SELECT** axis1 **ON COLUMNS**, axis2 **ON ROWS** **FROM** cube
FROM : une ou plusieurs tables	**FROM** : un cube
SELECT : • **une vue** des données en 2 dimensions : lignes (rows) et colonnes (columns)	**SELECT :** • **nombre quelconque de dimensions** pour former les résultats de la requête
Colonne : chaine de caractère ou valeur numérique	**Niveau :** Level
Plusieurs colonnes liées ou une table de dimension	**Dimension**

Exemple de requête MDX :	Signification
SELECT { [Time].[2003].[Q1], [Time].[2004].CHILDREN } **ON COLUMNS,** { [Markets].[APAC].[Australia], [Markets].[EMEA].[France] } **ON ROWS**	← Les colonnes : Q1 2003, Q1 2004, Q2 2004, Q3 2004, Q4 2004 ← Les lignes : Australia, France
FROM CubeSales	← Cube en question
WHERE ([MEASURES].[Sales], [Product].[Ships], [Customers].[All Customers])	← Agrégation de la mesure Sales avec la fonction SUM ← Sélection de la dimension Product (Ships seulement) ← Sélection de la dimension Customers (l'ensemble des clients)

SELECT : Description des axes du cube résultat	Chaque **dimension** du résultat est : • associée à un **rôle** correspondant à sa représentation dans le tableau retourné par la requête MDX • Exemples : ON COLUMNS, ON ROWS • sur un ou plusieurs **niveaux de l'hiérarchie** : • Exemple 1 : **{[Markets].[APAC].[Australia], [Markets].[EMEA].[France]}** de la dimension **Markets**, niveau **Pays** • Exemple 2 : **{[Time].[2003].[Q1], [Time].[2004].[Q1].CHILDREN}** de la dimension Temps, niveaux trimestre et mois (tous les mois du trimestre)
FROM : Spécification du/des cube/s de départ	• Ensemble de **cubes** nécessaires à la création du **cube résultat** • Si plusieurs cubes nécessaires, cela implique une **jointure multidimensionnelle** : chaque paire de cubes doit alors posséder au moins une dimension concordante
WHERE : Restriction sur le/s cube/s de départ	• **Restrictions sur le/s cube/s de départ** de la clause FROM • Spécification des restrictions par une **liste de noeuds de la hiérarchie d'une dimension** nommée **slicer-dimension**

Data Warehouse permet de : Transformer un système d'information en un système décisionnel Le Data Warehouse regroupe des données qui sont : orientées sujet , intégrées, non volatiles et datées La table de faits contient : Les mesures,Les mesures et les clés associées aux dimensions,Un nombre de colonnes réduit La table de dimension contient : Les détails sur les faits, Moins d'enregistrement qu'une table de faits, Un nombre de colonnes élevé Le modèle en étoile de Data Warehouse : Facilite la navigation, Centré sur la table de faits, Propose un nombre limité de jointures Le modèle en flocon de neige de Data Warehouse : Contient des tables de dimensions normalisées, Permet de réduire le volume de stockage des tables Le modèle en constellation de Data Warehouse : Permet le partage de dimension entre les tables de faits, Accepte plusieurs tables de faits, Peut contenir des tables de dimensions dé-normalisées Le processus ETL : Signifie Extraction – Transformation – Load, Permet de transformer les données acquises, Permet de charger dans une base de données cible des données traitées Le processus ETL est basé sur : Des connecteurs ,Des connecteurs et des transformateurs ,Des connecteurs, des transformateurs et des mise en correspondance





IDE de Talend L'IDE de développement est basé sur Eclipse et des connecteurs standards existent pour la majorité des sources de données, tels que les ERP, les BDD ou encore des solutions de commerces en ligne.
Capacité de Talend Talend est capable de transformer les données vers presque tous les formats existants et si une transformation ou un connecteur vous manque, il suffit de le développer très facilement en JAVA.
Point fort de Talend On aime donc sa facilité d'utilisation et sa polyvalence, mais attention la maitrise de JAVA est quand même souvent nécessaire pour comprendre les erreurs lors du développement.