Noël NOVELLI (noel.novelli@univ-amu.fr) ; Université d'Aix-Marseille; Département d'Informatique et Interactions et LIS (équipe DIAMS)

Intégration des Données

Partie 2 : Intégration centralisée et entrepôts de données

Deux familles d'Intégration des données

- Accès uniforme à plusieurs sources de données
  - Le passage à l'échelle (plusieurs sources de données)
  - L'hétérogénéité des (schémas) des sources
  - L'autonomie des sources
- Médiateurs (O. Boucelma)
  - Tout reste décentralisé, le médiateur transformera les données à la demande
- Entrepôt de données (N. Novelli)
  - Centralisé à un seul endroit (tout ou partie), la transformation et bien plus, sont faites avant l'utilisation des données

Intégration **centralisée** des données

- Accès uniforme à plusieurs sources de données
  - Le passage à l'échelle (plusieurs sources de données)
  - L'hétérogénéité des (schémas) des sources
  - L'autonomie des sources
- Entrepôt de données (N. Novelli)
  - Centralisé à un seul endroit (tout ou partie), la transformation est faite avant l'utilisation des données
- Prérequis
  - Diagramme de Classe, MCD et SQL

Intégration **centralisée** des données

- Objectifs
  - Centralisation des données provenant de plusieurs sources de données hétérogènes en un seul lieu
  - Préparation des données pour les exploiter le plus efficacement possible
    - Nettoyer et uniformiser les données
  - Enrichissement de l'information
    - Calculer tout ce qui peut l'être et utile lors de l'alimentation
    - Multidimensionnel et aide à la décision
  - Création d'entrepôts de données
- Problématiques
  - Stockage très volumineux des données et sélectionner les données à exploiter
  - Contrôler le rafraichissement des données
  - Contrôler la qualité de données

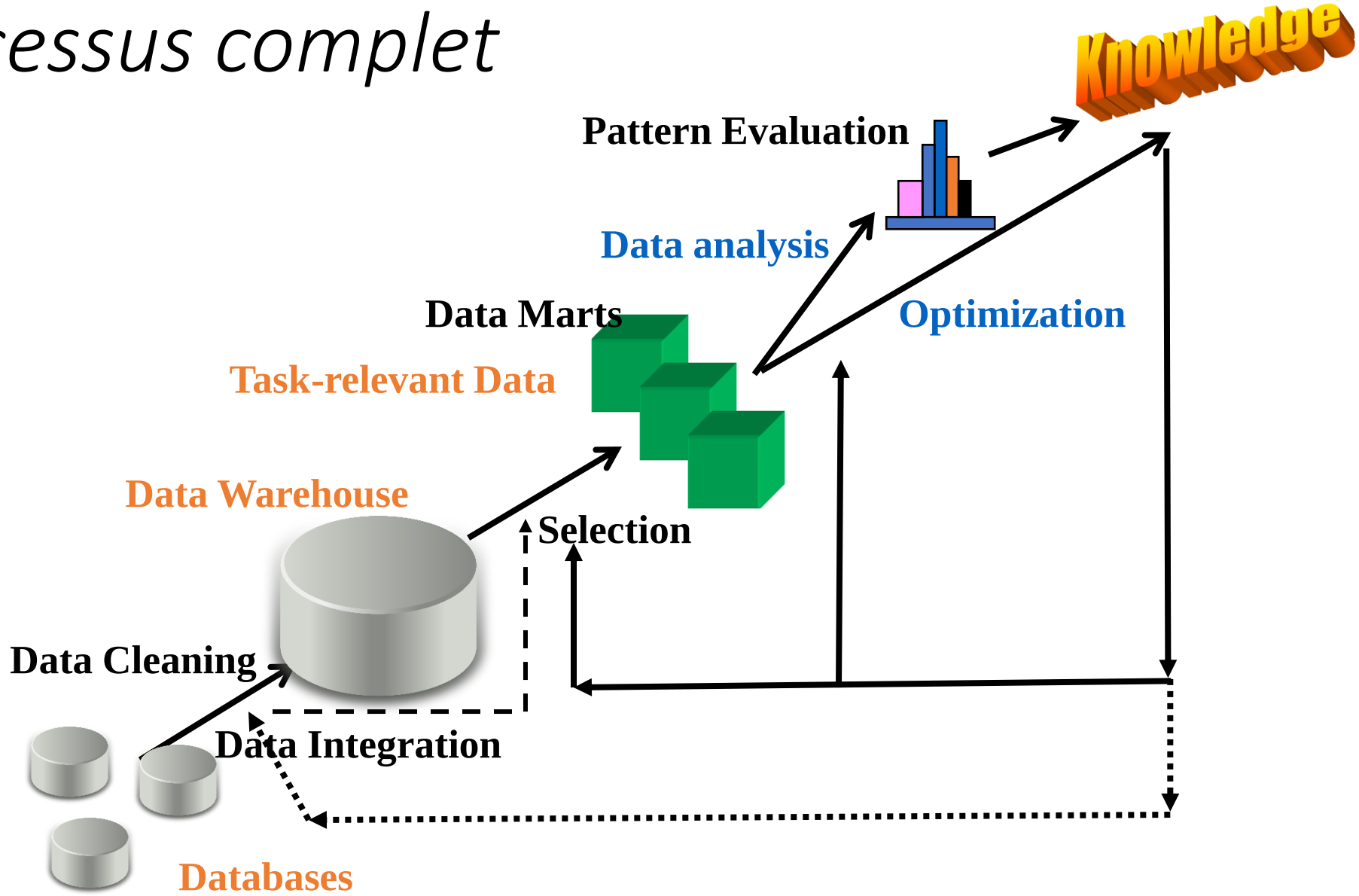
Intégration dans un entrepôt de données *Quelques définitions*

- Différentes définitions pas très rigoureuses
  - Une BD d'aide à la décision qui est maintenue séparément de la base opérationnelle de l'organisation
- « Un data warehouse est une collection de données concernant un sujet particulier, varie dans le temps, non volatile et où les données sont intégrées. »—W. H. Inmon
- Data warehousing
  - Le processus qui permet de construire un data warehouse

Intégration dans un entrepôt de données *Système d'Information d'Aide à la Décision*

- Qu'est ce qu'un SIAD ?
  - Des données
  - Des processus
  - Des mesures et des indicateurs associés
  - Les enjeux et objectifs
    - Externe (votre client) : optimisation, nouvelles connaissances, conquérir de nouveaux marchés, réduire des coûts…
    - Interne (votre entreprise) : acquérir un nouveau client ou fidéliser un client, accroître vos compétences, diversifications de vos clients…
  - Décideur : personne qui prendra des décisions basées sur vos rapports, tableaux de bord…

Intégration dans un entrepôt de données
Processus complet



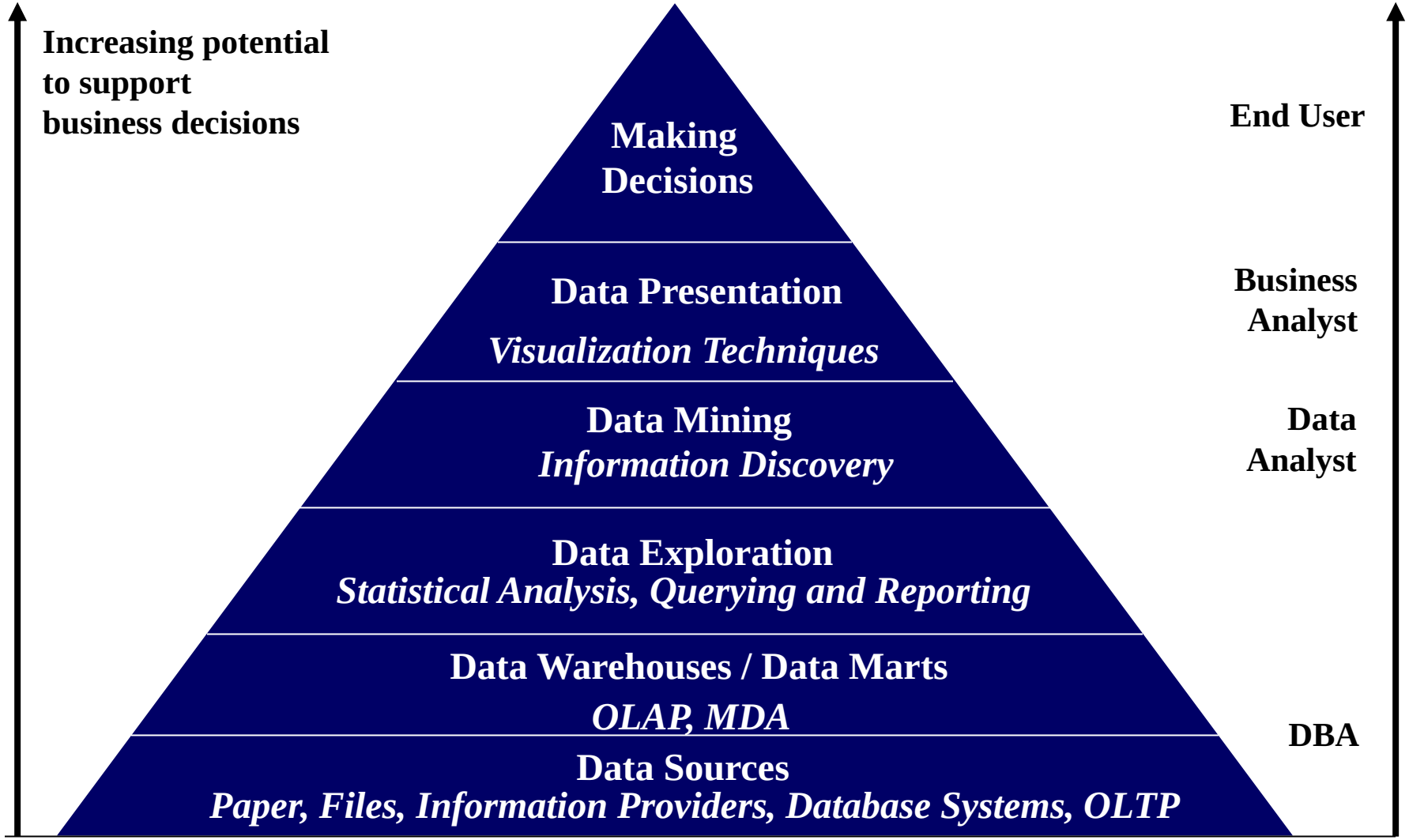
Recueil des sources de données *et leurs métadonnées*

- Où se trouvent-elles ? (URL, chemin d'accès sur le serveur…)
- Comment y accède-t-on ? (login/mdp, anonyme…)
- Quels sont leurs formats ? (XML, JSON, SGBD, Tableur…)
- Quelles sont leurs tailles ? (plusieurs Mo, Go, Po…)
- Quelle est la fréquence de rafraichissement ? (pour chaque source)
- Quelle est la fiabilité de ces données ? (réputations des sources)
- Quelles sont les différentes granularités présentes ? (heures, minutes…)
- Quels sont les axes d'analyse possible ? (temps, ventes, régions…)

Intégration dans un entrepôt de données *Choix des processus*

- En fonction de l'objectif : point de vue fonctionnel
- Coût des traitements : point de vue technique (consommation de ressources)
  - Regroupements (clustering)
  - Anomalies (outliers)
  - Corrélations (data dependencies)
  - Prédictions (what's next?)
  - …

Intégration dans un entrepôt de données
Système d'aide à la décision



Intégration dans un entrepôt de données
OLTP vs OLAP

	On Line Transaction Processing (OLTP)	**On Line Analytical Processing (OLAP)**
Utilisateurs	Naïf, Professionnel	Décideur
Fonction	Opérations journalières	Prise de décision
Conception BD	Orienté application	Orienté sujet d'étude
Données	Courantes, mises à jour, détaillées, données à plat, isolées	Historisées, résumées, multidimensionnelles, intégrées, consolidées
Utilisation	Répétitive	Spécifique
Accès	Lecture/Ecriture, Index/hash sur les clés	Nombreuses lectures
Session de travail	Courtes, transactions simples	Interrogations complexes
Nombre d'accès	Dizaine	Million
Nombre d'utilisateur	Milliers	Centaines
Taille des BD	100Mo-100Go	100Go-100To/Po
Performance	Débit transactionnel	Débit de réponse

Intégration dans un entrepôt de données *Orienté Sujet*

- Organisée autour d'un sujet bien précis
  - Exemple : client, produit, ventes, etc.
- S'intéresse à la modélisation et l'analyse des données pour aider les décideurs (OLAP), pas pour des activités quotidiennes ou traitement transactionnel (OLTP)
- Fournit une vue simple et concise concernant un sujet particulier en excluant les données qui ne servent pas à la prise de décision

Intégration dans un entrepôt de données
Données intégrées

- Construite en intégrant plusieurs sources de données potentiellement hétérogènes
    - BD relationnelles, BD NoSQL, Web, fichiers plats…
- Les techniques d'intégration et de nettoyage des données sont utilisées
    - Garantir la consistance des conventions de nommage (les attributs `Nom` et `Nom_Famille` dans `BD1` et `BD2` désignent la même chose)
    - Structures de codage (l'attribut `Nom` est sur 15 `char` et 20 `char` sur `BD1` et `BD2`; `NSS` est une chaîne dans `BD1` et c'est un entier long dans `BD2`)
    - Domaines des attributs (exemple : `cm` vs `pouce`), etc.
    - C'est au moment où les données sont copiées dans le data warehouse qu'elles sont uniformisées

Intégration dans un entrepôt de données
Varie dans le temps

- La portée temporelle des données dans un data warehouse est plus longue que celle des bases opérationnelles
  - Base opérationnelle : valeur courante des données
  - Data warehouse : fournit des infos sous une perspective historique
    - Exemple : 5 à 10 dernières années d'information
- Dans un data warehouse, en général, chaque donnée fait référence au temps mais dans une base opérationnelle les données peuvent ne pas faire référence au temps, il faudra donc ajouter la notion de temps aux données intégrées.

Intégration dans un entrepôt de données *Data Warehouse est Non-Volatile*

- Un support de stockage séparé des sources de données
- Les mises à jour de la base opérationnelle n'ont pas lieu au niveau du data warehouse
  - N'a pas besoin de modules de gestion de transactions (concurrence, reprise sur panne…)
  - N'a besoin que de deux opérations pour accéder aux données :
    - Chargement initial des données (écriture) et interrogation (lecture)
    - Pas de mis à jour des données, historisation des données

Intégration dans un entrepôt de données *Data Warehouse vs. SGBD hétérogènes*

- Traditionnellement, l'intégration de BD hétérogènes se fait par le biais de :
  - Wrappers/médiateurs au dessus des BD hétérogènes *(cf. Partie 1 du cours, O. Boucelma)*
  - Approche orientée requête
    - Quand une requête est posée par un site client, un méta-dictionnaire est utilisé pour la traduire en plusieurs requêtes appropriées à chacune des BD source. Le résultat est l'intégration des réponses partielles
    - L'exécution des requêtes demande donc beaucoup de ressources
- Data warehouse : Approche orientée mise à jour
  - Les infos sont intégrées et stockées pour une interrogation directe. Plus efficace en coût d'exécution des requêtes mais stockage d'un grand volume de données

Intégration dans un entrepôt de données *Data Warehouse vs. BD Opérationnelle*

- OLTP (On-Line Transaction Processing)
  - Exécution en temps réel des transactions, pour l' enregistrement des opérations quotidiennes : inventaire, commandes, paye, comptabilité…
  - Par opposition aux traitements en batch
- OLAP (On-Line Analytical Processing)
  - Traitement efficace des requêtes d'analyse pour la prise de décision qui sont par défaut assez complexes (même si elles peuvent être réalisées par les SGBDR classiques)
- OLTP vs. OLAP
  - …

Intégration dans un entrepôt de données *Data Warehouse vs. BD Opérationnelle*

- OLTP (On-Line Transaction Processing)
  - …
- OLAP (On-Line Analytical Processing)
  - …
- OLTP vs. OLAP
  - Données : courantes, détaillées *vs* historiques, consolidées
  - Conception : modèle ER + application *vs* Modèle en étoile + sujet
  - Vue : courante, locale *vs* évolutive, intégrée
  - Modes d'accès : mise à jour *vs* lecture seule mais requêtes complexes

Intégration dans un entrepôt de données *Pourquoi pas des BD comme data warehouses*

- Les 2 systèmes sont performants
  - SGBD calibrés pour l'OLTP : méthodes d'accès, index, contrôle de concurrence, reprise
  - Warehouse calibrés pour l'OLAP : requêtes OLAP complexes, vue multidimensionnelle, consolidation.
- Fonctions et données différentes
  - Données manquantes : l'aide à la décision a besoin des données historiques qui ne se trouvent pas dans les BD opérationnelles
  - Consolidation : l'AD a besoin de données consolidées (agrégats) alors qu'elles sont brutes dans les BD opérationnelles

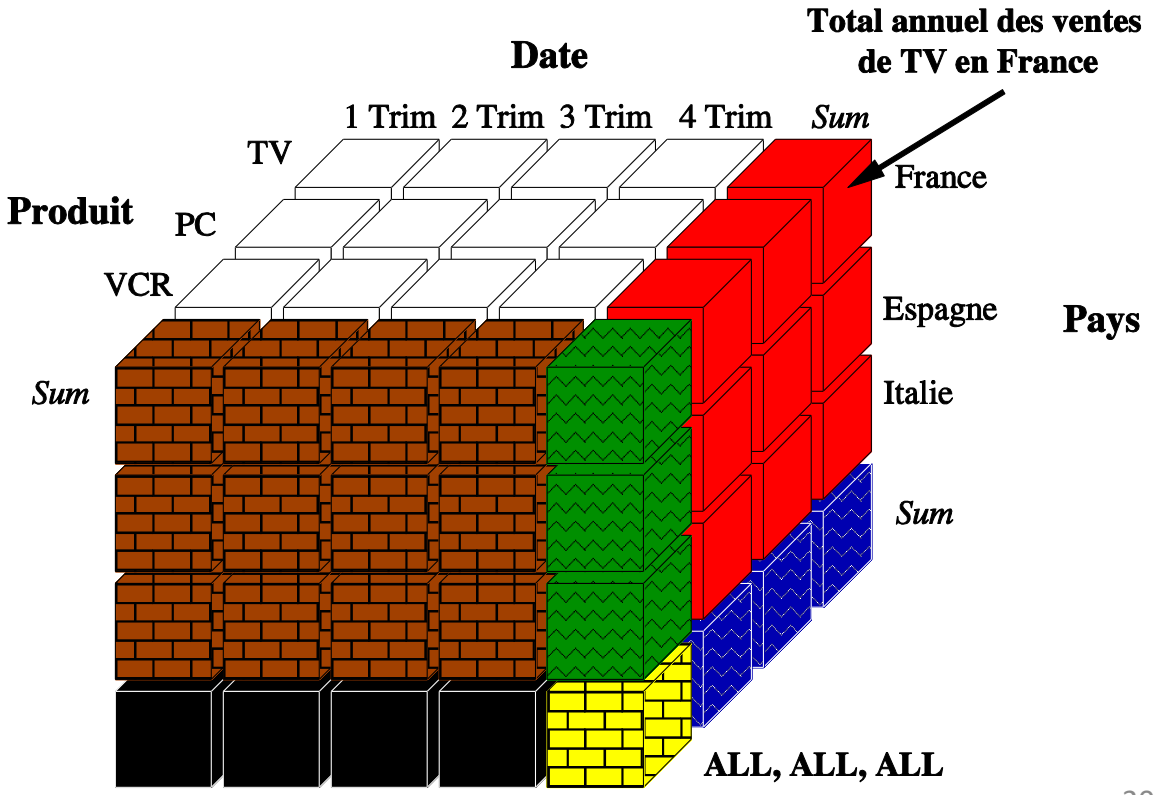
Intégration dans un entrepôt de données
Des *Tables aux Data cubes*

- Un data warehouse est basé sur un modèle multidimensionnel où les données sont vues comme des data cubes
- Un data cube permet de voir les données selon plusieurs dimensions
  - Exemple sales
    - Les tables de dimension : item (nom_item, marque, type), ou temps(jour, semaine, mois, trimestre, année)
    - La table de faits contient des mesures (unités_vendues) et les clés externes faisant référence à chaque table de dimension
- Dans la littérature du data warehousing, un cube de dimension *n* est dit un cuboïde. Le treillis des cuboïdes d'un data warehouse forme un data cube.

Intégration dans un entrepôt de données Des *Cubes de Données (1/3)*

- Pré-calcul du résultat des requêtes OLAP
- Généralisation du Group By de SQL

SELECT Produit, Date, Pays, **SUM**(Quantité)
FROM Vente
GROUP BY CUBE(Produit, Date, Pays);



Intégration dans un entrepôt de données Des *Cubes de Données (2/3)*

- Pré-calcul du résultat des requêtes OLAP
- Généralisation du Group By de SQL

SELECT A, B, C, SUM(M) FROM Table GROUP BY CUBE(A, B, C)

⇨

SELECT All, All, All, SUM(M) FROM Table UNION
SELECT **A**, All, All, SUM(M) FROM Table Group By **A** UNION
SELECT All, **B**, All, SUM(M) FROM Table Group By **B** UNION
SELECT All, All, **C,** SUM(M) FROM Table Group By **C** UNION
SELECT **A**, **B**, All, SUM(M) FROM Table Group By **A**, **B** UNION
SELECT **A**, All, **C,** SUM(M) FROM Table Group By **A**, **C** UNION
SELECT All, **B**, **C,** SUM(M) FROM Table Group By **B**, **C** UNION
SELECT **A**, **B**, **C,** SUM(M) FROM Table Group By **A**, **B**, **C**

Intégration dans un entrepôt de données
Des *Cubes de Données (3/3)*

Données initiales			
Vendeur	Date	Produit	Ventes
1	d1	a	10
1	d2	a	9
1	d3	a	8
1	d1	b	6
1	d2	b	10
1	d3	b	15
2	d1	a	10
2	d2	a	19
2	d3	a	81
2	d1	b	5
2	d2	b	10
2	d3	b	5

SELECT V, D, P FROM Ventes GROUP BY Cube(V, D, P)					
Dimensions	∑ ventes	Dimensions	∑ ventes	Dimensions	∑ ventes
∅	188	2,d2	29	1,d1,a	10
1	58	2,d3	86	1,d2,a	9
2	130	1,a	27	1,d3,a	8
d1	31	1,b	31	1,d1,b	6
d2	48	2,a	110	1,d2,b	10
d3	109	2,b	20	1,d3,b	1
a	137	d1,a	20	2,d1,a	10
b	51	d1,b	11	2,d2,a	19
1,d1	16	d2,a	28	2,d3,a	81
1,d2	19	d2,b	20	2,d1,b	5
1,d3	23	d3,a	89	2,d2,b	10
2,d1	15	d3,b	20	2,d3,b	5

Quel est le meilleur vendeur, à quelle date et quel produit ?

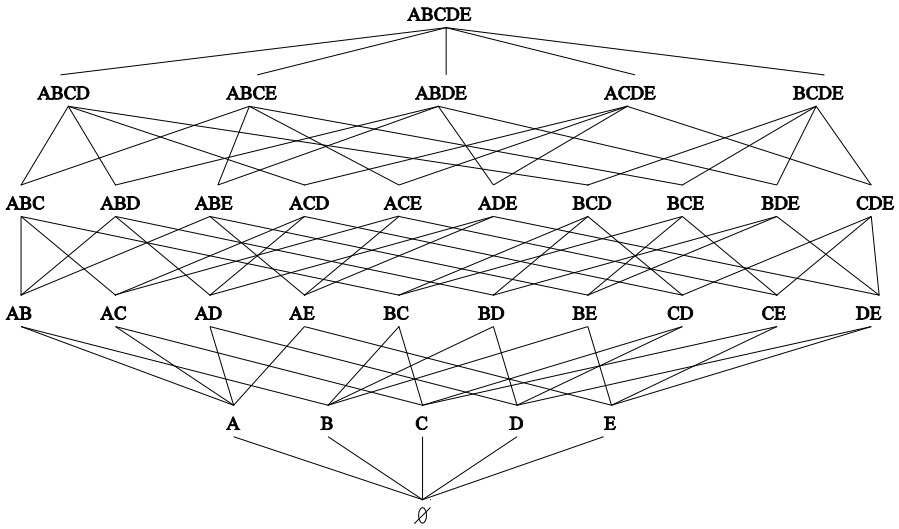
Intégration dans un entrepôt de données
Des *Cube de données : Problématique (1/2)*

Agréger les valeurs mesures, à l'aide d'une fonction *f*, suivant toutes les combinaisons de dimensions possibles (cube complet), ou sous certaines conditions (cube incomplet)

Pour une relation *r* de schéma
$\{Dim, M\}$ où $Dim = \{A_1, A_2, \ldots, A_k\}$,
$\forall X \subseteq Dim$, calculer *f(M)*

- *Espace de recherche : ensemble des parties de Dim*
- *Calculer f(M) nécessite des accès disque*

Ecriture sur disque : $2^k \leq \sum_{d \subseteq Dim} \left(\prod_{i=1}^{|d|} \|d\| \right) \leq n2^k$

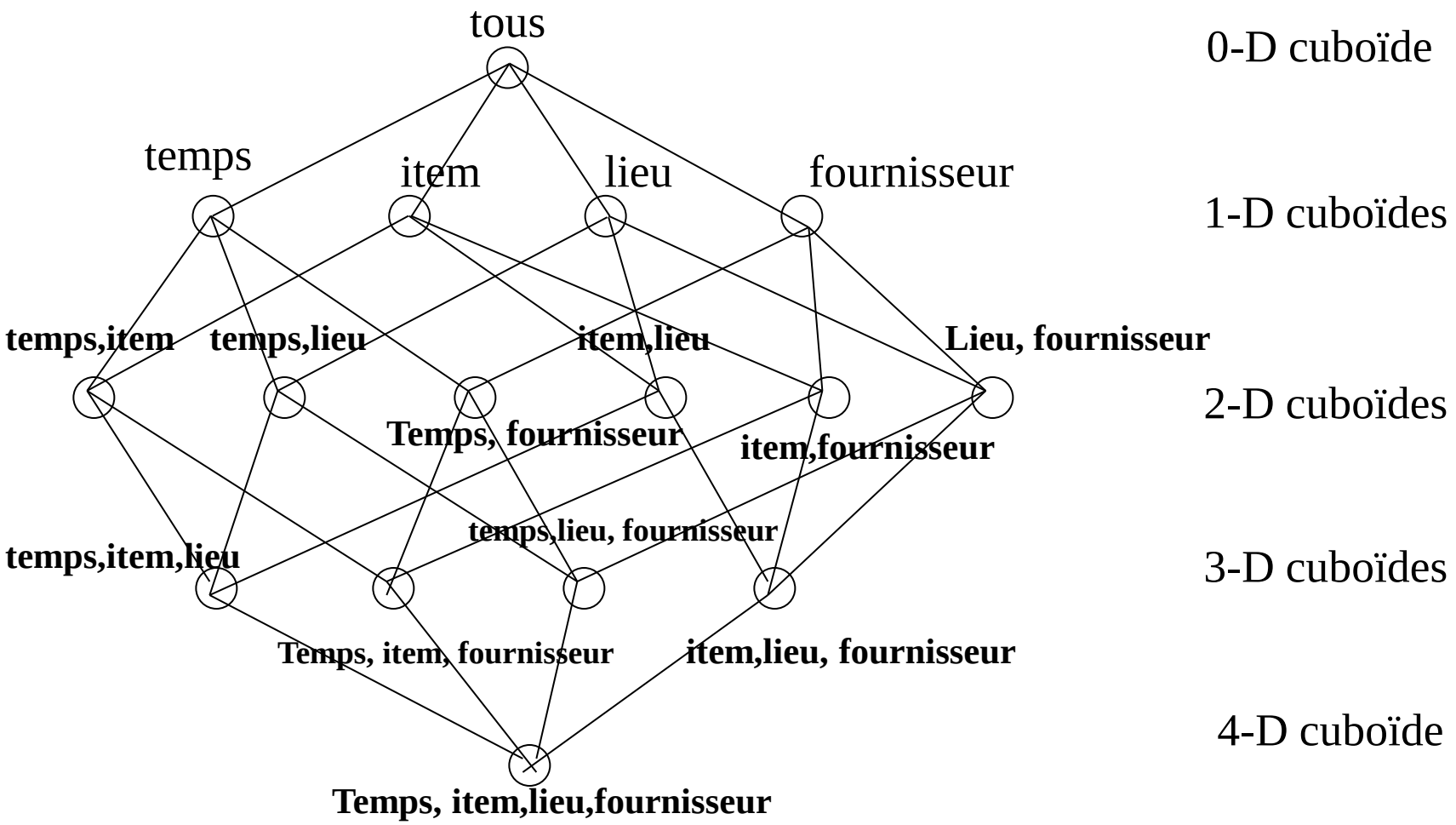


Treillis des parties de { A, B, C, D, E }

Intégration dans un entrepôt de données
Des *Cube de données : Problématique (2/2)*

- Parcourir l'ensemble des parties de *Dim* (exponentiel)
- Minimiser les accès disque
- Minimiser la mémoire centrale nécessaire
- Eparpillement des données (lié aux deux points précédents)
- Explosion disque, stockage du cube de données (plusieurs Giga à Peta)
  - Technique de compression de données
  - Modèle d'approximation (inadapté aux données éparpillées)
  - Sélection de vues à matérialiser

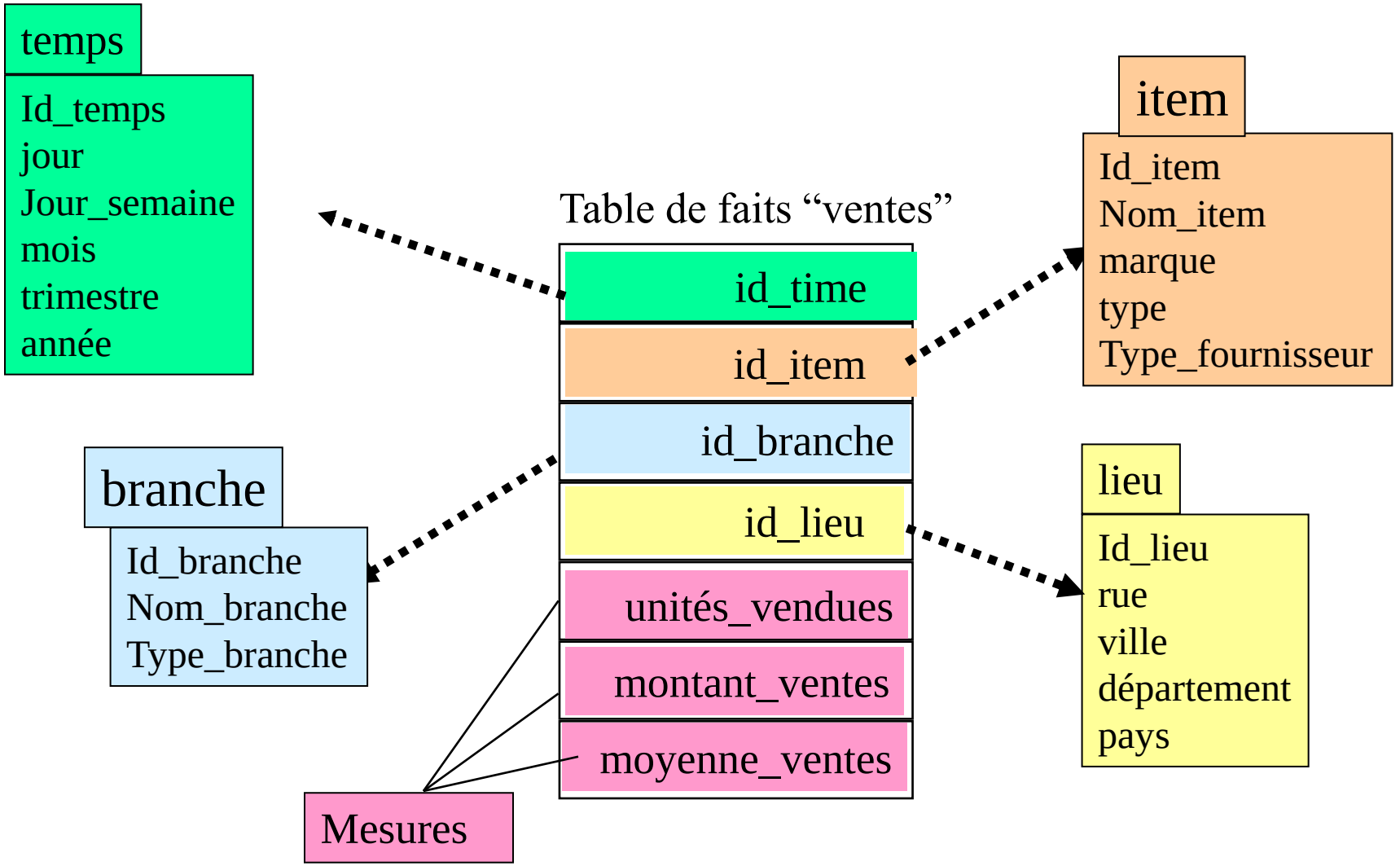
Intégration dans un entrepôt de données Des *Cube : Un treillis de cuboïdes*



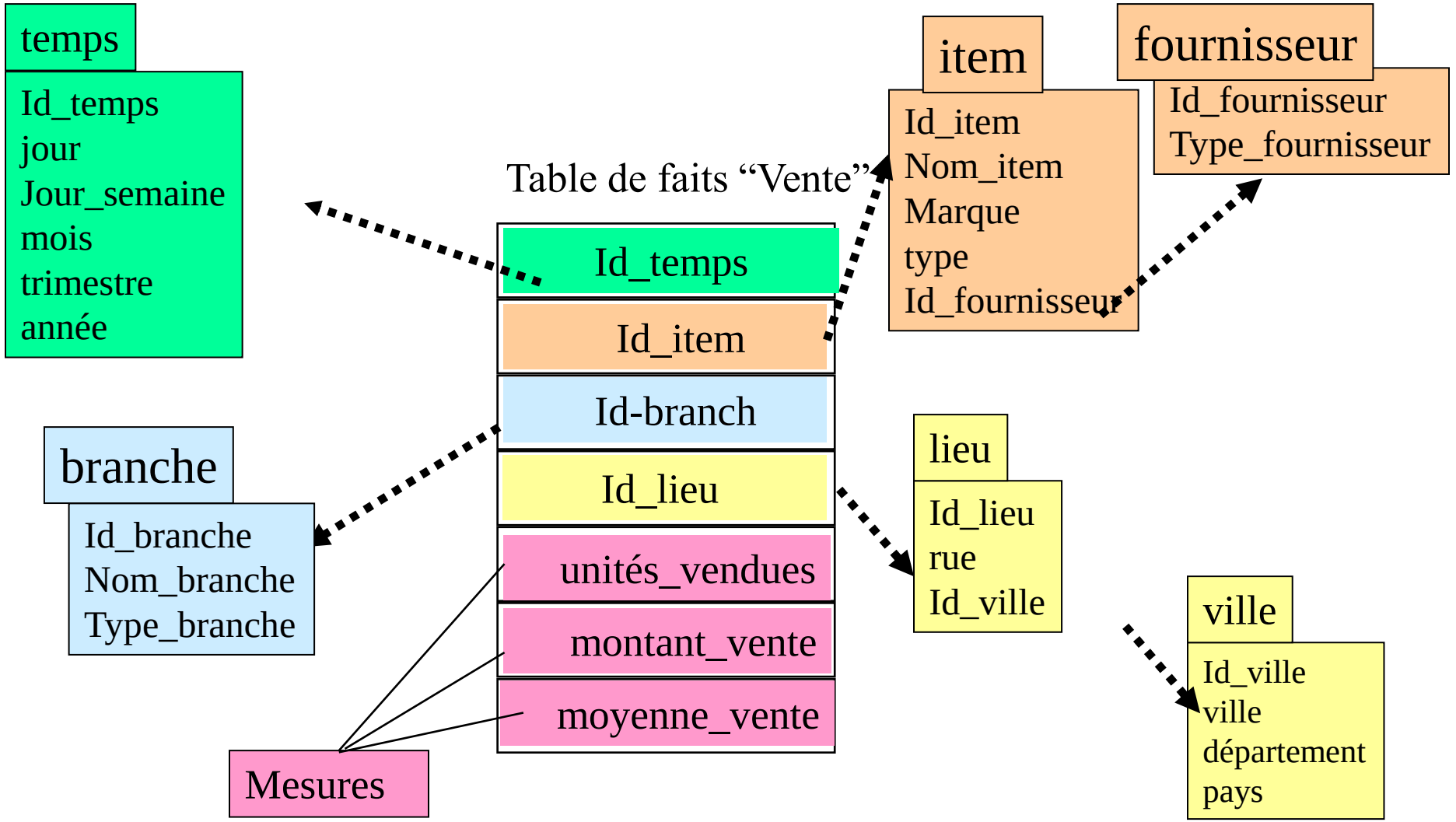
Intégration dans un entrepôt de données *Modélisation Conceptuelle*

- Dimensions & mesures
  - Schéma en étoile : une table de faits au centre connectée à un ensemble de tables de dimensions
  - Schéma flocon de neige : un raffinement du précédent où certaines tables de dimensions sont normalisées, décomposées
  - Constellation de faits : plusieurs tables de faits partageant au moins une table de dimension (constellation d'étoiles et/ou de flocons)

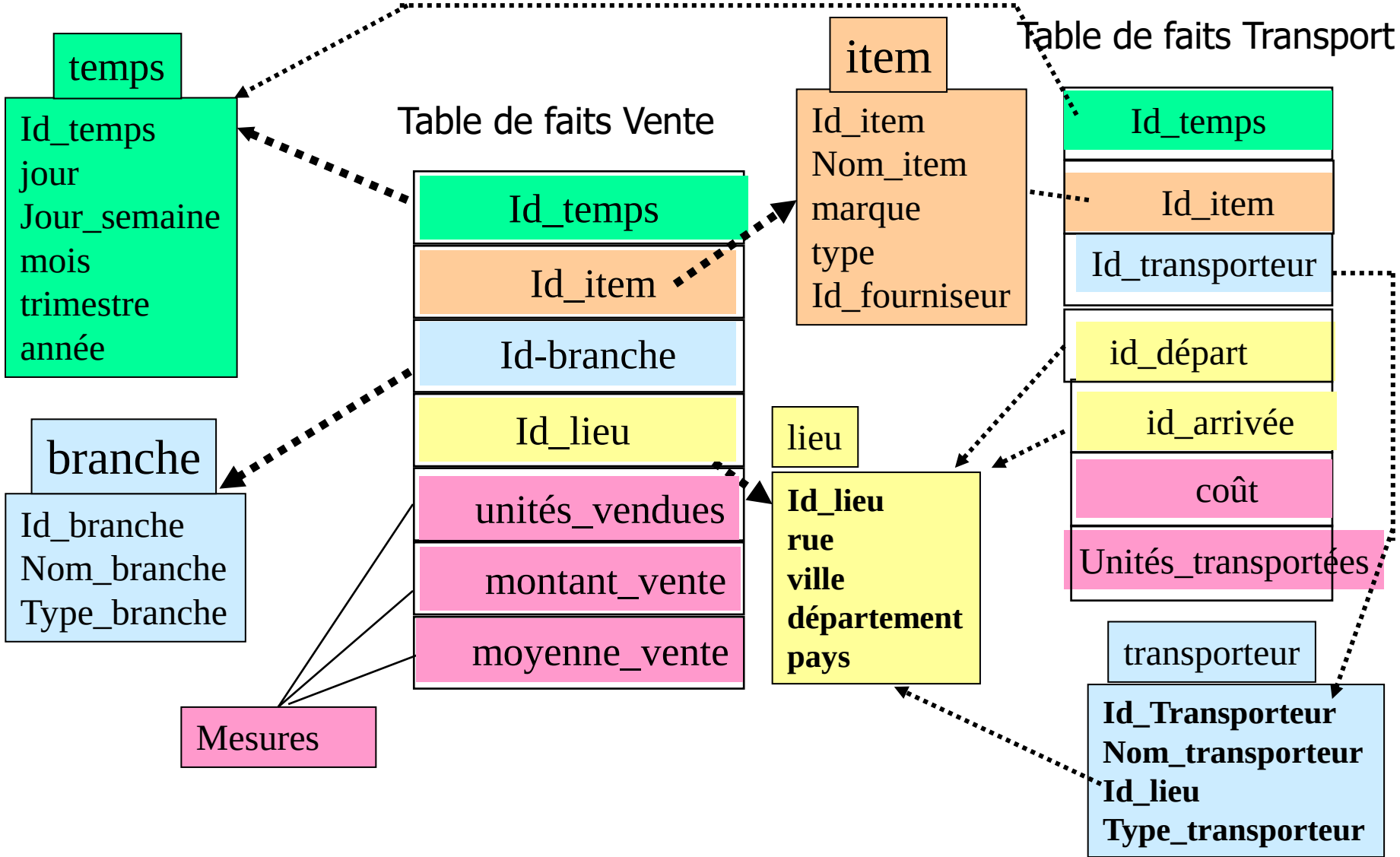
Intégration dans un entrepôt de données
Schéma en étoile



Intégration dans un entrepôt de données
Schéma en flocon



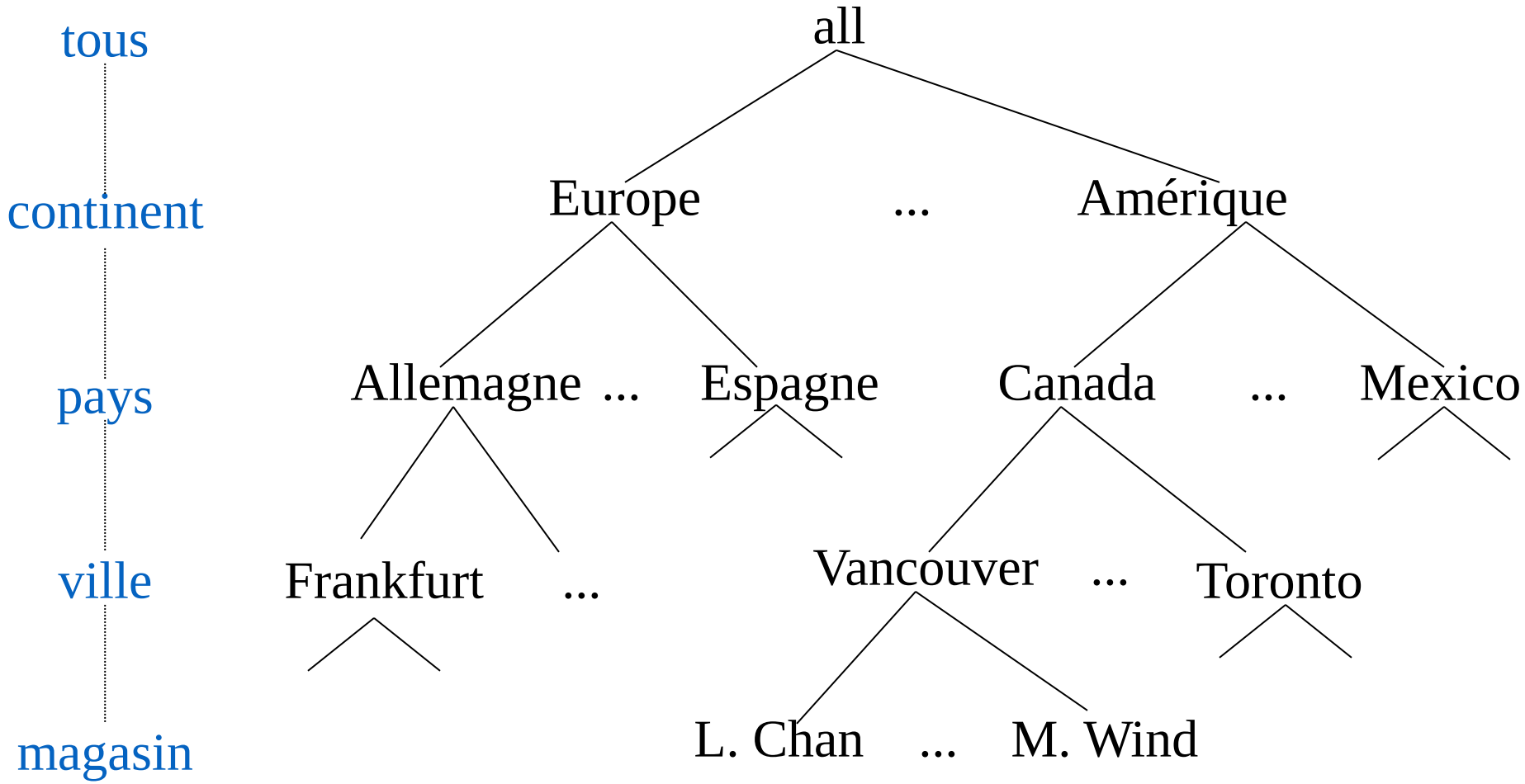
Intégration dans un entrepôt de données
Schéma en constellation



Intégration dans un entrepôt de données *Mesures : Trois Catégories*

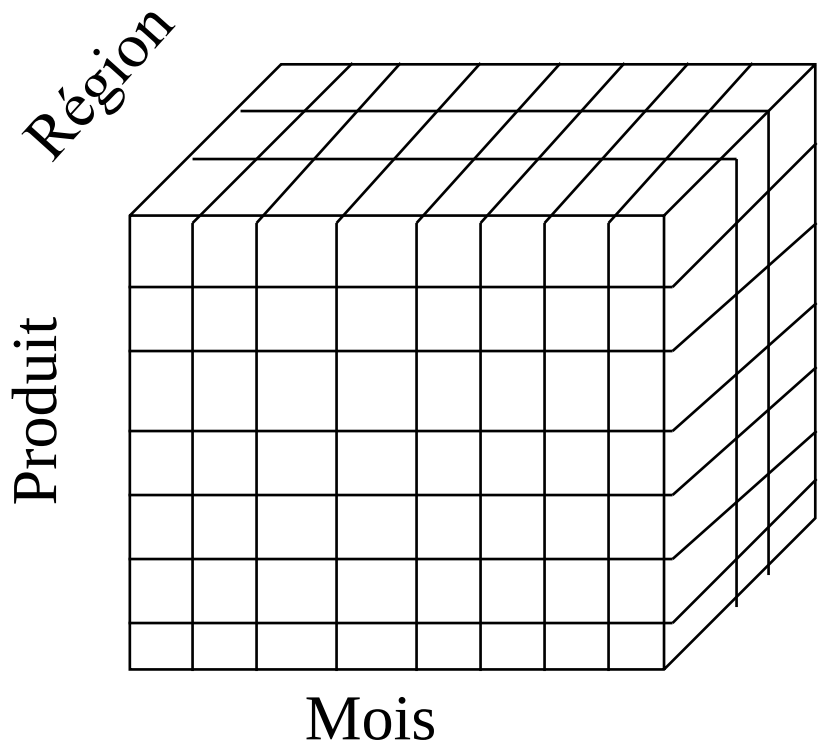
- Distributives/Additives : le résultat obtenu par une fonction à *N* valeurs calculées est le même que le résultat de la fonction sur toutes les valeurs.
    - Exemple : count(), sum(), min(), max()
- Algébriques : elle peut être calculée par une fonction à *M* arguments chacun obtenu par une fonction distributive.
    - Exemple : avg(), min_N(), standard_deviation()
- Holistique : le résultat ne peut pas être obtenu par les valeurs précédemment calculées, il faut tout recalculer à chaque nouvelle valeur.
    - Exemple : median(), mode(), rank()

Intégration dans un entrepôt de données
Hiérarchie de la Dimension (exemple : Lieu)

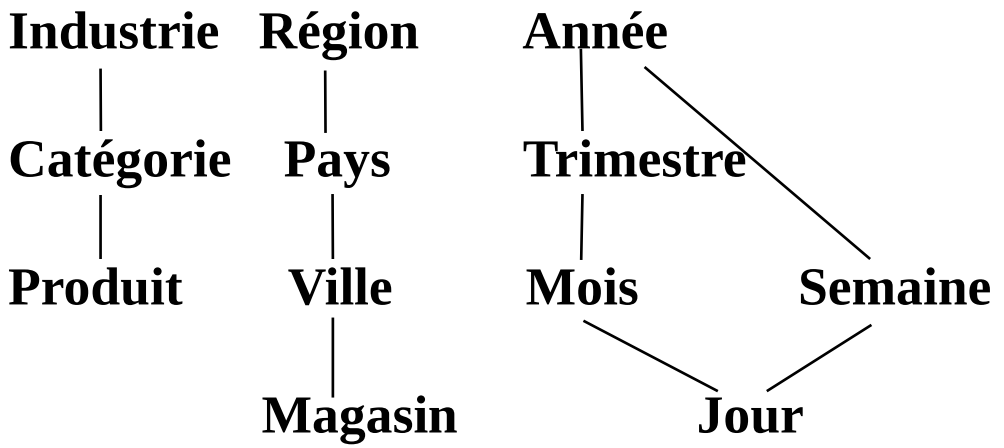


Intégration dans un entrepôt de données *Données multidimensionnelles*

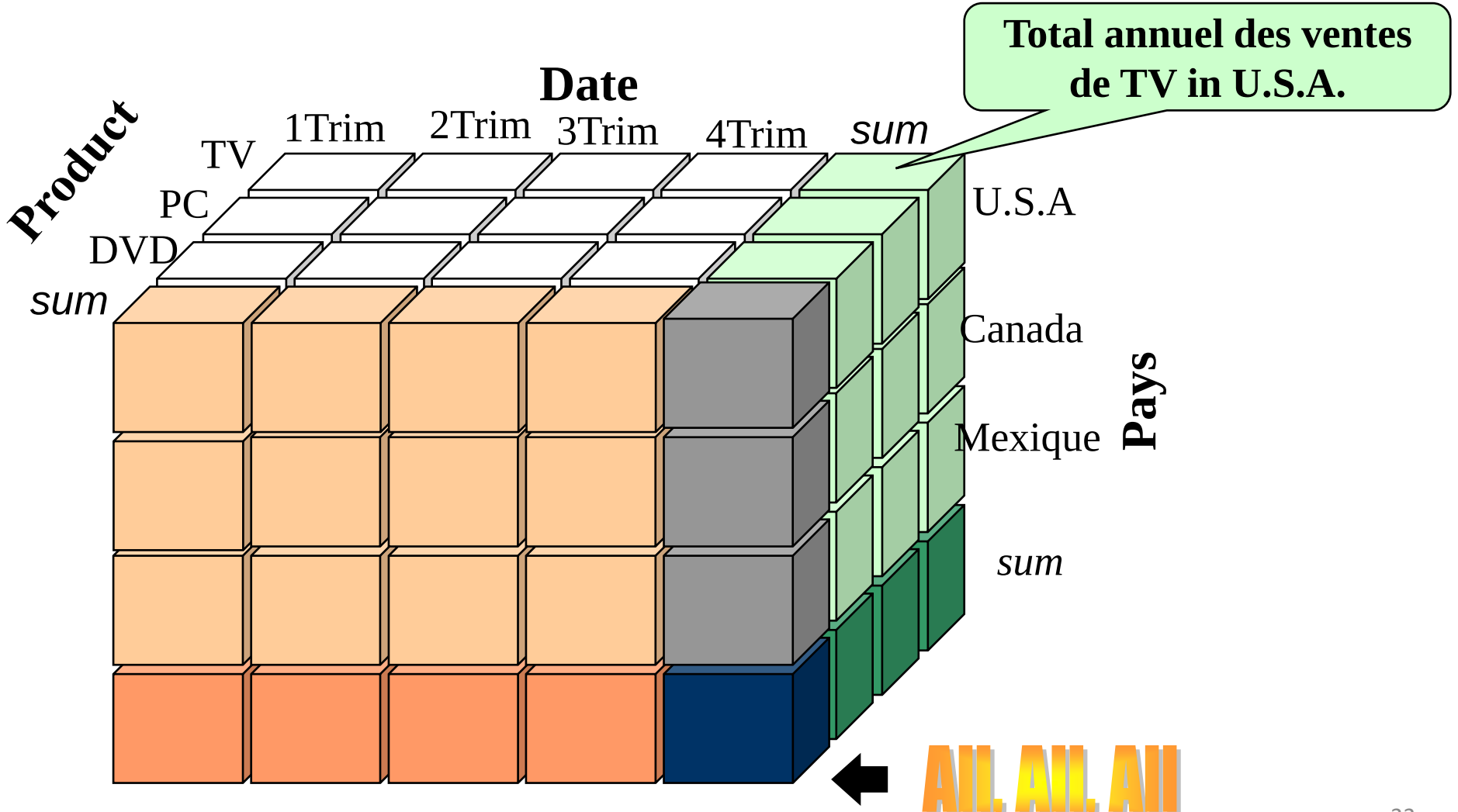
- Montant des ventes comme une fonction des paramètres produit, mois, région



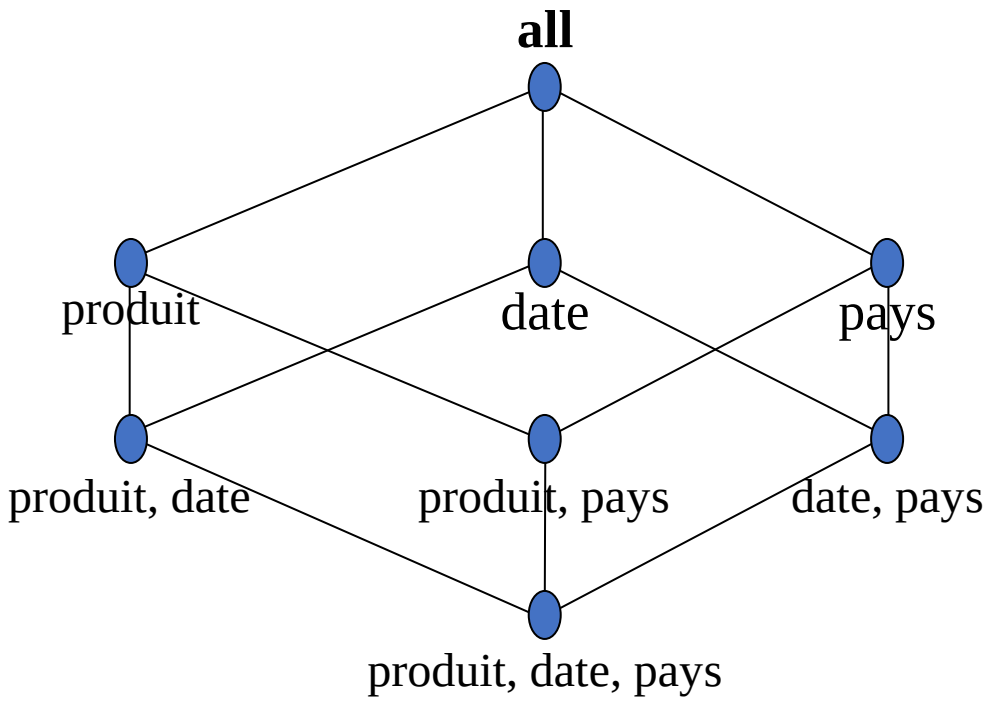
Dimensions : Produit, Lieu, Temps
Chemins de consolidation hiérarchiques



Intégration dans un entrepôt de données
Données multidimensionnelles



Intégration dans un entrepôt de données
Cuboïdes Correspondants au Cube



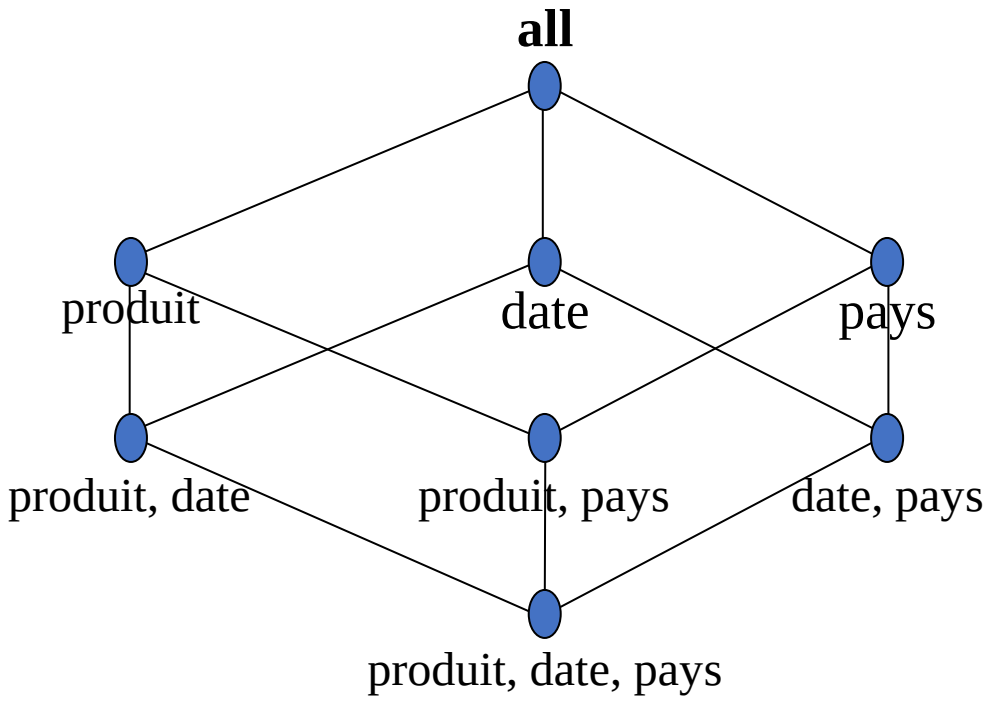
Cuboïde 0-D(apex)

Cuboïde 1-D

Cuboïde 2-D

Cuboïde3-D (base complète)

Intégration dans un entrepôt de données
Naviguer dans un cube de données



Cuboïde 0-D(apex)

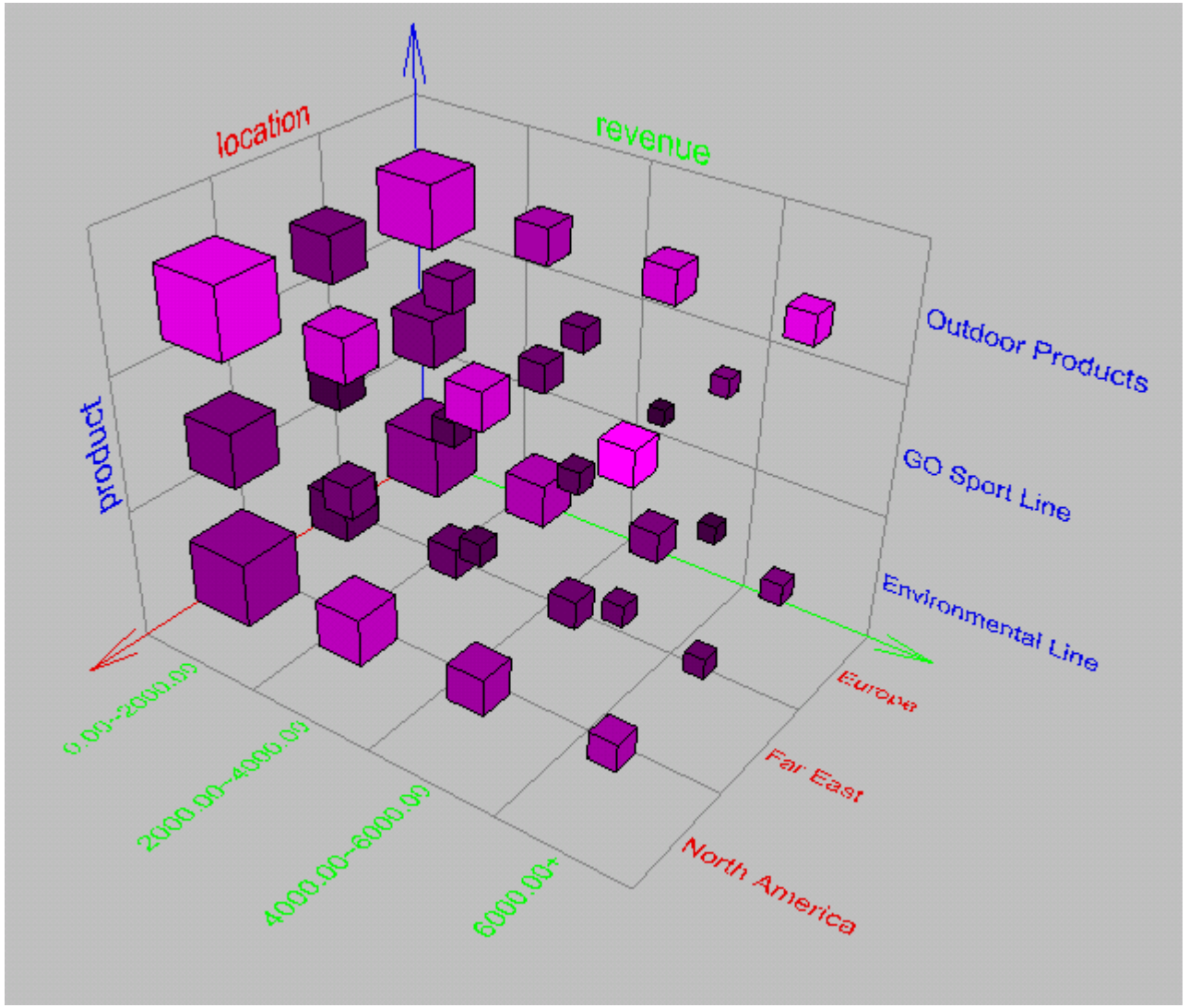
Cuboïde 1-D

Cuboïde 2-D

Cuboïde3-D (base complète)

Intégration dans un entrepôt de données
Naviguer dans un cube de données

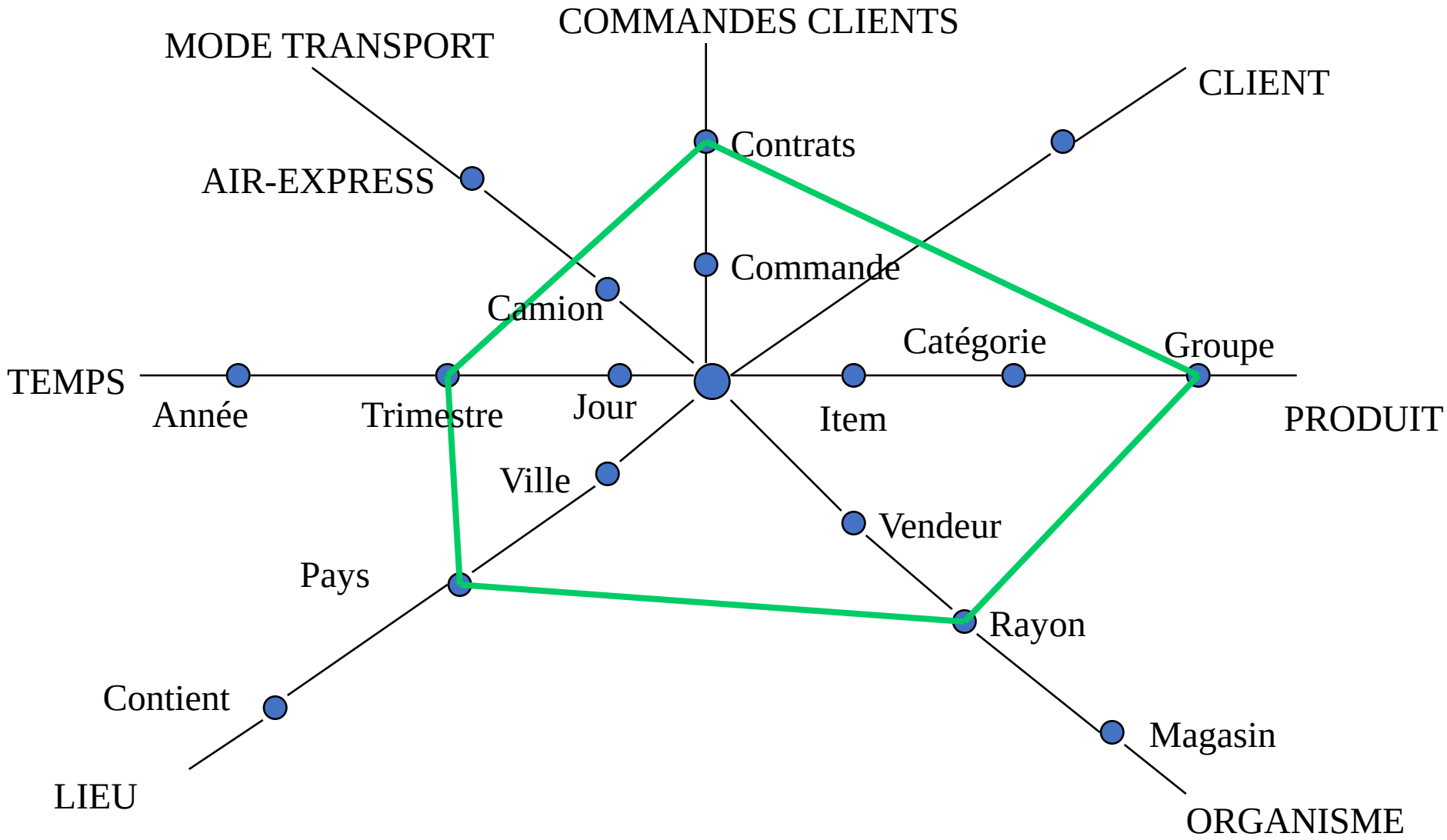
- Visualisation
- Opérations OLAP
- Manipulation interactive



Intégration dans un entrepôt de données
Opérations typiques de l'OLAP

- Roll up : consolider (résumer) les données
  - *Passer à un niveau supérieur dans la hiérarchie d'une dimension*
- Drill down : l'inverse du Roll up
  - *Descendre dans la hiérarchie d'une dimension*
- Slice et Dice :
  - *Projection et sélection du modèle relationnel*
- Pivot (rotate) :
  - *Orientation du cube pour visualisation*

Intégration dans un entrepôt de données
Un Modèle pour représenter les requêtes



Architecture d'un data warehouse
Différentes vues de l'analyse

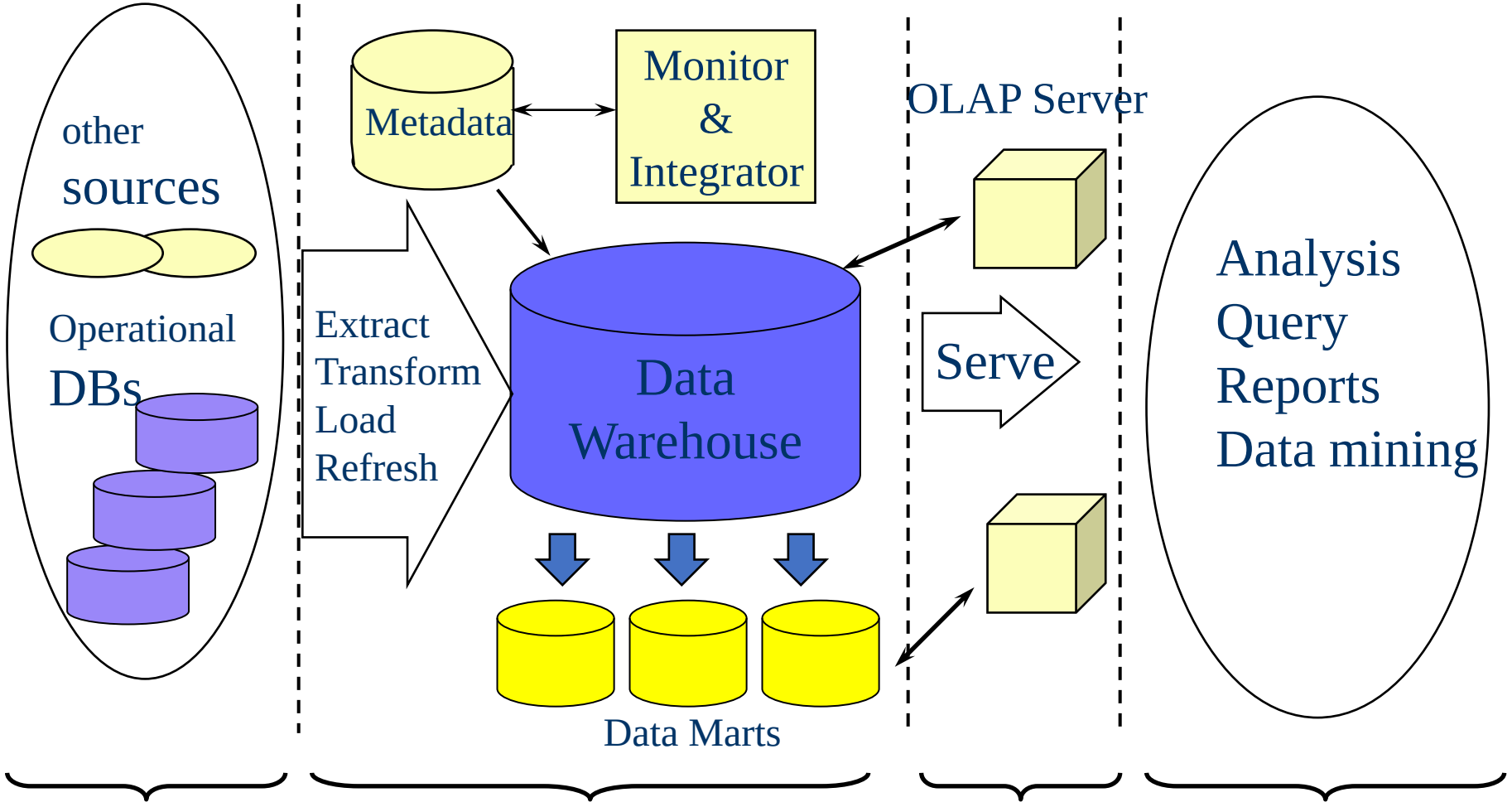
Quatre façons de regarder la conception d'un entrepôt de données

- Vue Top-down
  - Sélectionner les informations pertinentes nécessaires à l'entrepôt de données
- Vue Data source
  - Déterminer les informations à récupérer, conserver et gérer avec le système opérationnel
- Vue Data warehouse
  - Constituer de la ou les tables de faits et des tables de dimensions
- Vue Business query
  - Montre les possibilités des données qu'offre l'entrepôt de données pour les utilisateurs finaux

Architecture d'un data warehouse
Choix du processus de modélisation

- Approche top-down, bottom-up ou les deux
  - Top-down : du plus général aux plus détaillé (mature)
  - Bottom-up : des expérimentations aux prototypes (rapide)
- D'un point vue génie logiciel
  - Cascade ou V : méthode structurée et systématique, un pas après l'autre
  - Spirale : méthode agile, itérative par incrémentation de fonctions, cycle court
- Processus classique de conception d'un data wharehouse
  - Choisir le sujet d'étude à modéliser, par exemple : commandes, factures, etc.
  - Choisir la granularité des données pour le processus
  - Choisir les dimensions utilisées pour chaque table de faits
  - Choisir les mesures qui seront présentes dans la table de faits

Architecture d'un data warehouse
Architecture n-tiers



Architecture d'un data warehouse
Trois modèles de data warehouse

- Entreprise warehouse
  - Collecte de toutes les informations concernant les sujets traités au niveau de l'organisation
- Data Mart
  - Un sous ensemble d'un entreprise warehouse. Il est spécifique à un groupe d'utilisateurs. Par exemple, data mart du marketing.
- Data warehouse virtuel
  - Un ensemble de vues définies à partir de la base opérationnelle
  - Seulement un sous ensemble des vues sont matérialisées

Architecture d'un data warehouse
Serveur OLAP

- Relational OLAP (ROLAP)
  - Utilise un SGBD Relationnel pour stocker les données ainsi qu'un middle-ware pour implémenter les opérations spécifiques de l'OLAP si nécessaire. De plus en plus de SGBD Relationnel sont capable de faire des calcul OLAP optimisés.
- Multidimensional OLAP (MOLAP)
  - Basé sur un stockage par tableaux (techniques des matrices creuses)
  - Indexation rapide de données calculées
- Hybrid OLAP (HOLAP)
  - Souplesse pour l'utilisateur, i.e., bas niveau : relationnel, haut niveau : matrices creuses
- Serveurs SQL spécialisés
  - Support dédié aux requêtes SQL sur des schémas décisionnels

Architecture d'un data warehouse
Exemple de matérialisation partielle

- Comme données initiales, nous avons des informations sur des Produits (*p*) proposés par des Fournisseurs (*f*) et vendus à des Clients (*c*) à un prix (*PV*). Les informations s'étalent sur 10 ans.
- Les analystes voudraient poser des requêtes sur une table où chaque < *p, f, c* > est associé à une mesure *TV* (Total Ventes).
- Le produit *p* proposé par *f* a été vendu à *c* pour un montant global *TV* sur les 10 ans.

Architecture d'un data warehouse
Exemple de matérialisation partielle

- On considère un ensemble de requêtes, un ensemble de vues possibles
- Quelles vues matérialisées pour répondre à toutes les requêtes, si le nombre de vues ne doit pas dépasser un certain seuil ?
- Les requêtes considérées sont de la forme

SELECT <g-attributs>, SUM (<mesure>)
FROM <la table de base>
WHERE <attribut = valeur>
GROUP BY <g-attributs>;

⇨

SELECT Produit, Client, SUM(TV)
FROM Vente
WHERE Client='aClient'
GROUP BY (Produit, Client)

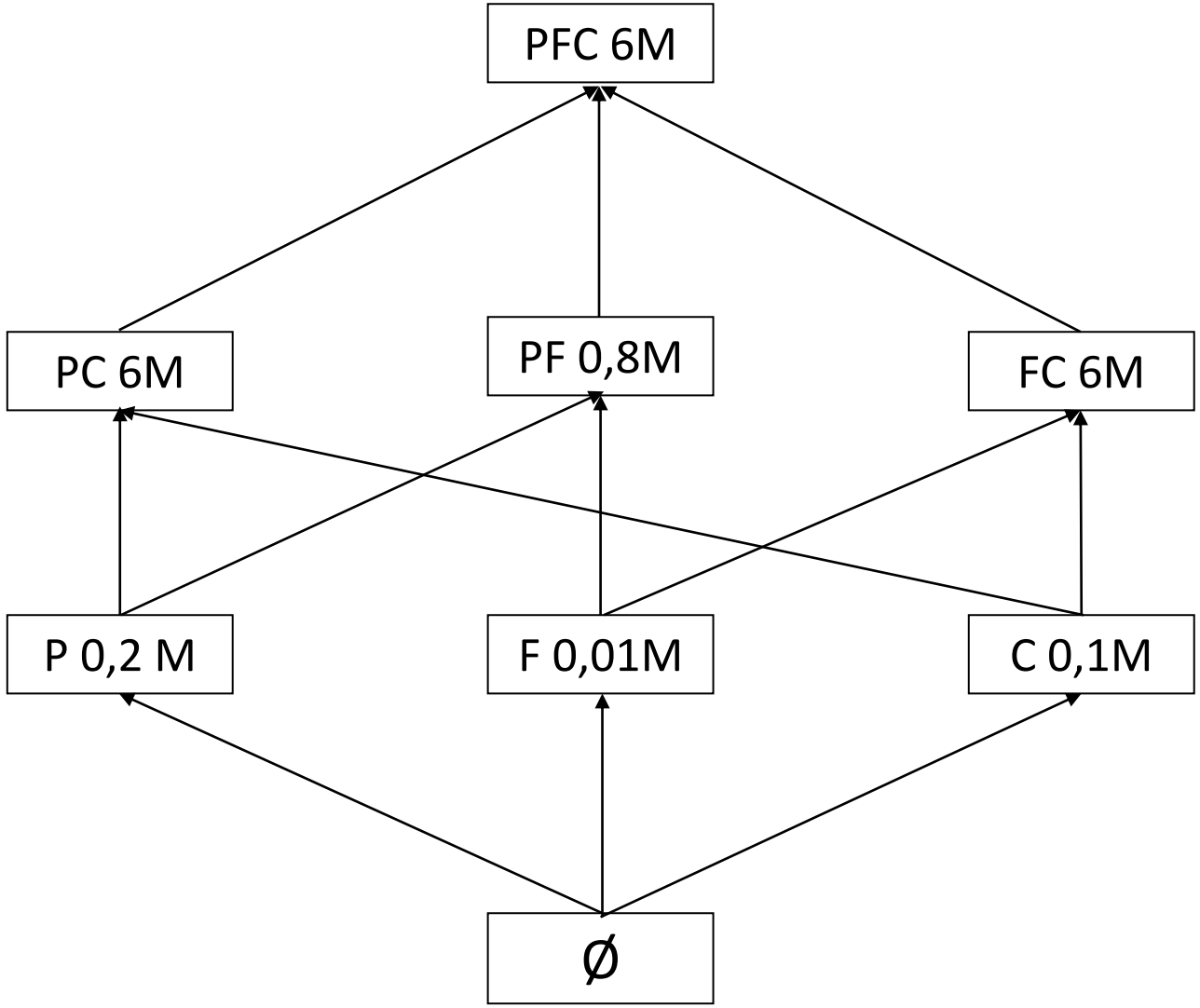
Architecture d'un data warehouse
Exemple de matérialisation partielle

- Les requêtes considérées sont de la forme

  SELECT <g-attributs>, SUM (<mesure>) **FROM** <la table de base>
  WHERE <attribut = valeur> **GROUP BY** <g-attributs>;

- Dans l'exemple, il y a 8 vues (matérialisation complète) :
  - Produit, fournisseur, client (6M tuples)
  - Produit, client (6M)
  - Produit, fournisseur (0,8M)
  - Fournisseur, Client (6M)
  - Produit (0,2M)
  - Fournisseur (0,01M)
  - Client (0,1M)
  - Ø (1)

Architecture d’un data warehouse
Exemple de matérialisation partielle



Nettoyage des données

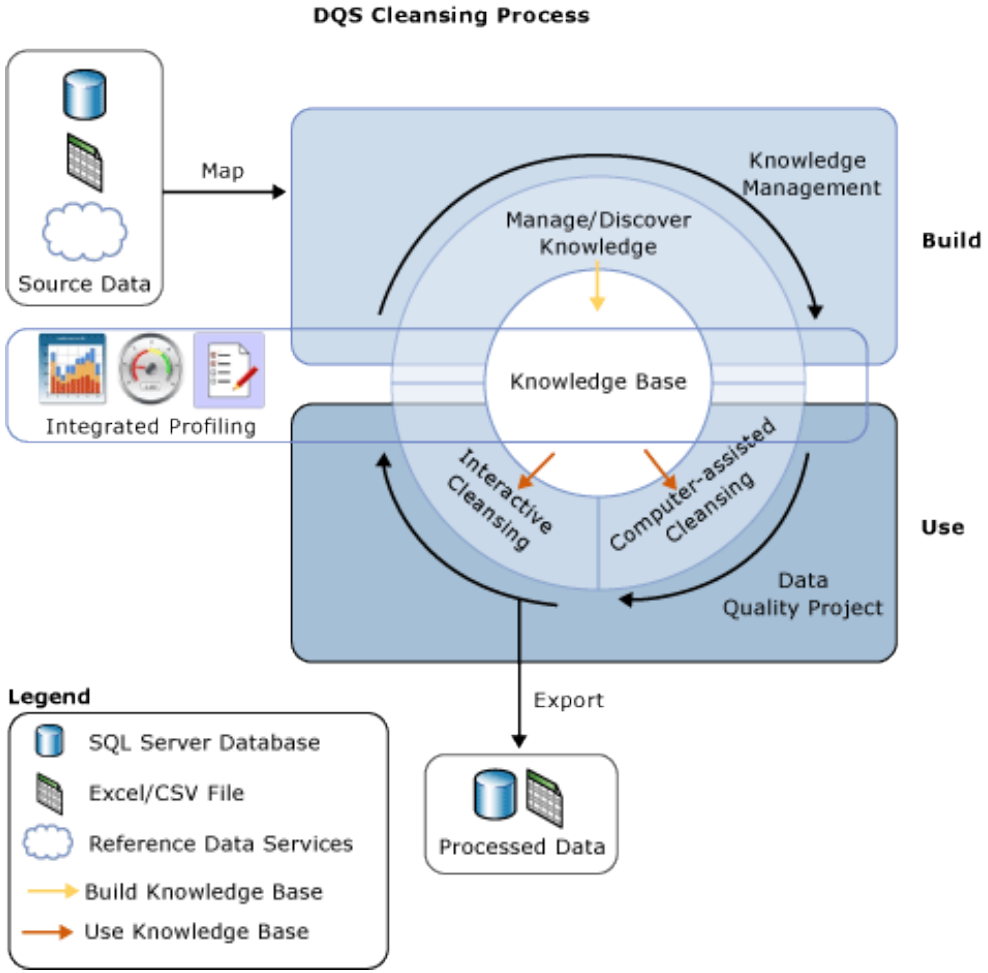
- Objectifs
  - Vérifier les données
  - Corriger les données
  - Standardiser, uniformiser les formats
  - Optimiser ou au moins améliorer la qualité des données

Nettoyage des données

- Données impropres
  - Données mal formatées
  - Données fausses, incorrectes, erronées
  - Données incohérentes
  - Données incomplètes, tronquées, censurées
  - Données manquantes ou déguisées
  - Données obsolètes
  - Données ambigües
  - Données en double, multiple exemplaires, etc.

Nettoyage des données
Base de connaissances

- Plusieurs techniques principales
  - Base de connaissances
    - Vérification et correction grâce à celle-ci
  - Reference ou Master Data
  - Analyse des Données



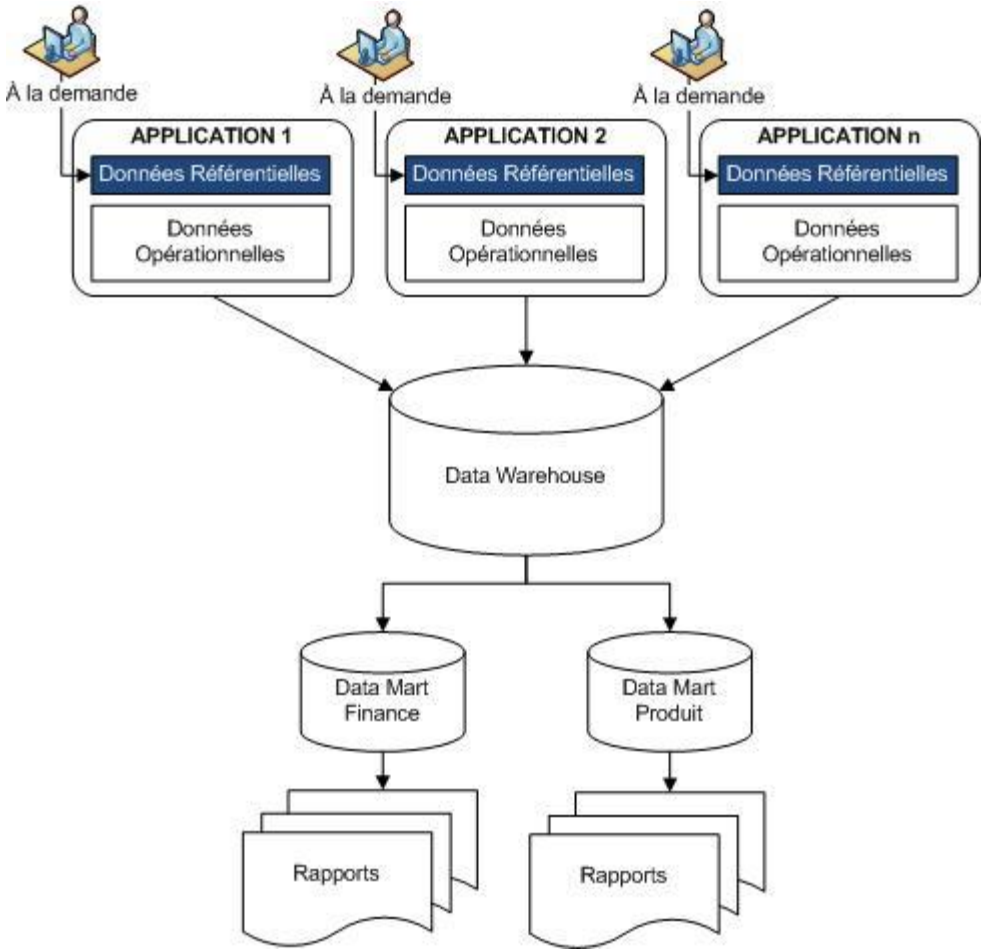
SQL Server 2012, Microsoft

Nettoyage des données
Base de connaissances

- Le nettoyage de données est le processus qui consiste à analyser la qualité des données contenues dans une source de données, à approuver/refuser manuellement les suggestions formulées par le système et à apporter les modifications correspondantes aux données. Le nettoyage de données dans Data Quality Services (DQS) inclut un processus assisté par ordinateur qui analyse la conformité des données par rapport aux connaissances contenues dans une base de connaissances et un processus interactif qui permet au gestionnaire de données d'examiner et de modifier les résultats du processus assisté par ordinateur afin de s'assurer que le nettoyage de données correspond exactement à ce qu'il souhaite faire.
- Le gestionnaire de données peut également procéder au nettoyage des données dans le cadre du processus de création de package d'Intégration Services. Dans ce cas, le gestionnaire de données utilise le composant de nettoyage DQS dans Intégration Services qui nettoie automatiquement les données à l'aide d'une base de connaissances existante. Pour plus d'informations, consultez Transformation de nettoyage DQS.
- La fonctionnalité de nettoyage de données de DQS présente les avantages suivants :
  - Identifie les données incomplètes ou incorrectes dans la source de données (fichier Excel ou base de données SQL Server), puis les corrige ou vous informe de la présence de données non valides.
  - Fournit un processus de nettoyage des données en deux étapes : *assisté par ordinateur* et *interactif*. Le processus assisté par ordinateur utilise les connaissances figurant dans une base de connaissances DQS pour traiter automatiquement les données et suggérer des remplacements/corrections. Le processus suivant est interactif. Il permet au gestionnaire de données d'approuver, de rejeter ou de modifier les modifications proposées par DQS au cours du nettoyage assisté par ordinateur.
  - Normalise et enrichit les données client à l'aide de valeurs de domaine, de règles de domaine et de données de référence .Par exemple, vous pouvez normaliser l'utilisation du terme « Street » de manière à remplacer « St. » par « Street » et enrichir les données en ajoutant les éléments manquants de manière à remplacer« 1 Microsoft way Redmond 98006 » par « 1 Microsoft Way, Redmond, WA 98006 ».
  - Fournit une interface de type Assistant à la fois simple, intuitive, et cohérente permettant à l'utilisateur de parcourir les données et d'examiner les erreurs dans un ensemble de données très volumineux.

Nettoyage des données *Master Data Management*

- Plusieurs techniques
  - Base de Connaissances
  - *reference data* ou *master data* : données déjà validées et consolidées (Ville dans un département par exemple), commune à tout le SI
    - Utilisation de standard, données validées et de qualité connue
    - Réduit les coûts et les erreurs de nomenclature, données validées et de qualité connue
  - Analyse des données

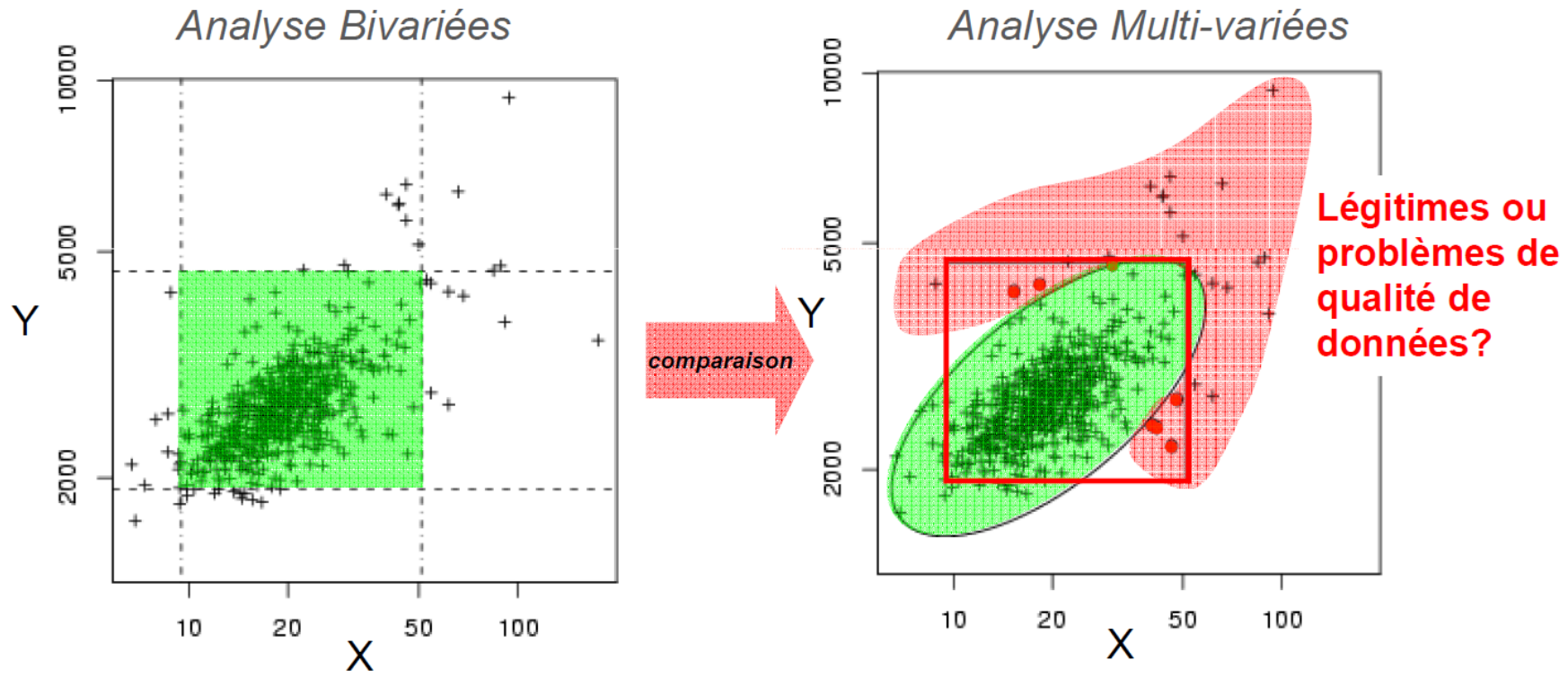


https://fr.wikipedia.org/wiki/Gestion_des_données_de_référence

Nettoyage des données
Analyse des Données

- Plusieurs techniques
  - Base de Connaissances
  - *reference data* ou *master data*
  - Analyse des données
    - Statistiques
    - Lois de distributions des données
    - Détection de valeurs par défaut
    - Détection *outliers*

Nettoyage des données
Outliers

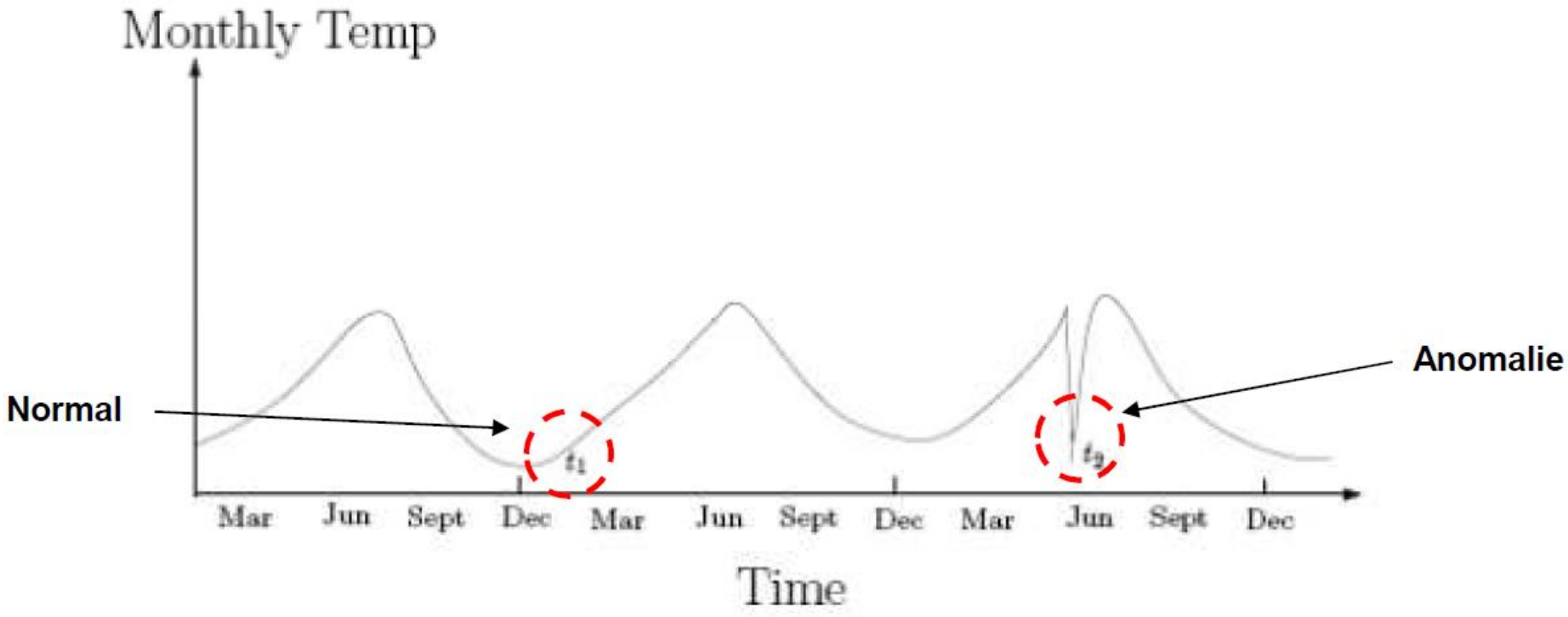


Aire de rejet en dehors des 2% et 98% sur X et Y

Rejet basé sur:
Mahalanobis_dist(cov(X,Y)) > $\chi^2(.98,2)$

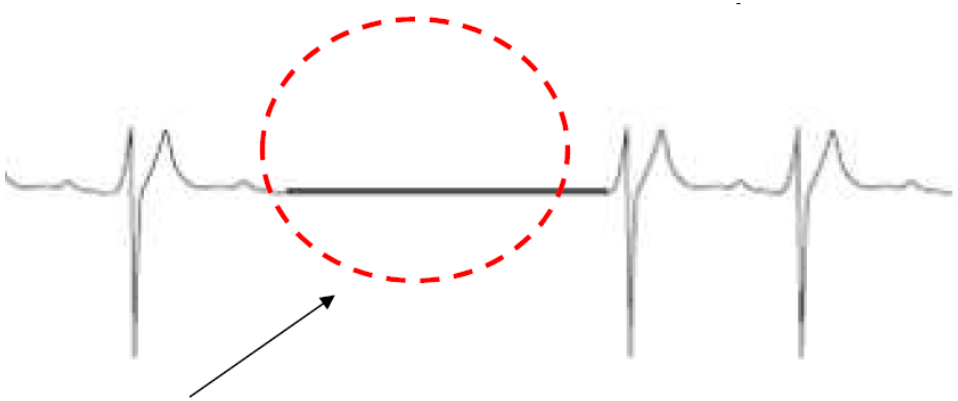
Nettoyage des données
Anomalies contextuelles

- observation qui dévie de la norme par rapport à un contexte



Nettoyage des données
Anomalies collectives

- Une collection d'observations anormales
- Nécessité de l'existence d'une relation entre les observations
  - Séquentialité
  - Spatialité
  - Connectivité (graphe)
- Chaque instance d'une anomalie collective n'est pas une anomalie en soi



Nettoyage des données
Taxonomies des méthodes de détection