



UNIVERSITE DE CORSE  
Master Informatique  
parcours DFS et DE

1<sup>ère</sup> année  
2025-2026

## BD partie 3 CH4 – Exploitation d'un DW

Evelyne VITTORI  
[vittori\\_e@univ-corse.fr](mailto:vittori_e@univ-corse.fr)

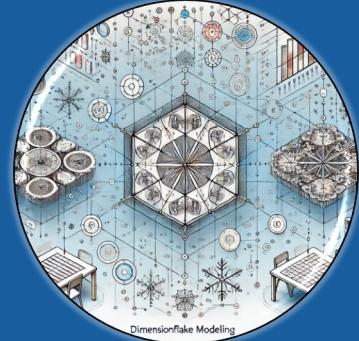


# Plan du cours



## CH1 – Principes des Datawarehouse

- Objectifs
- Différences avec une BD
- Architectures DW, DL et DLH



## CH2 – Modélisation dimensionnelle

- Concepts de modélisation dimensionnelle
- Schémas en étoile et en flocon



## CH3 – Processus ETL

- Définition et rôle d'un processus ETL
- Principaux outils
- Mise en pratique avec PentahoDI

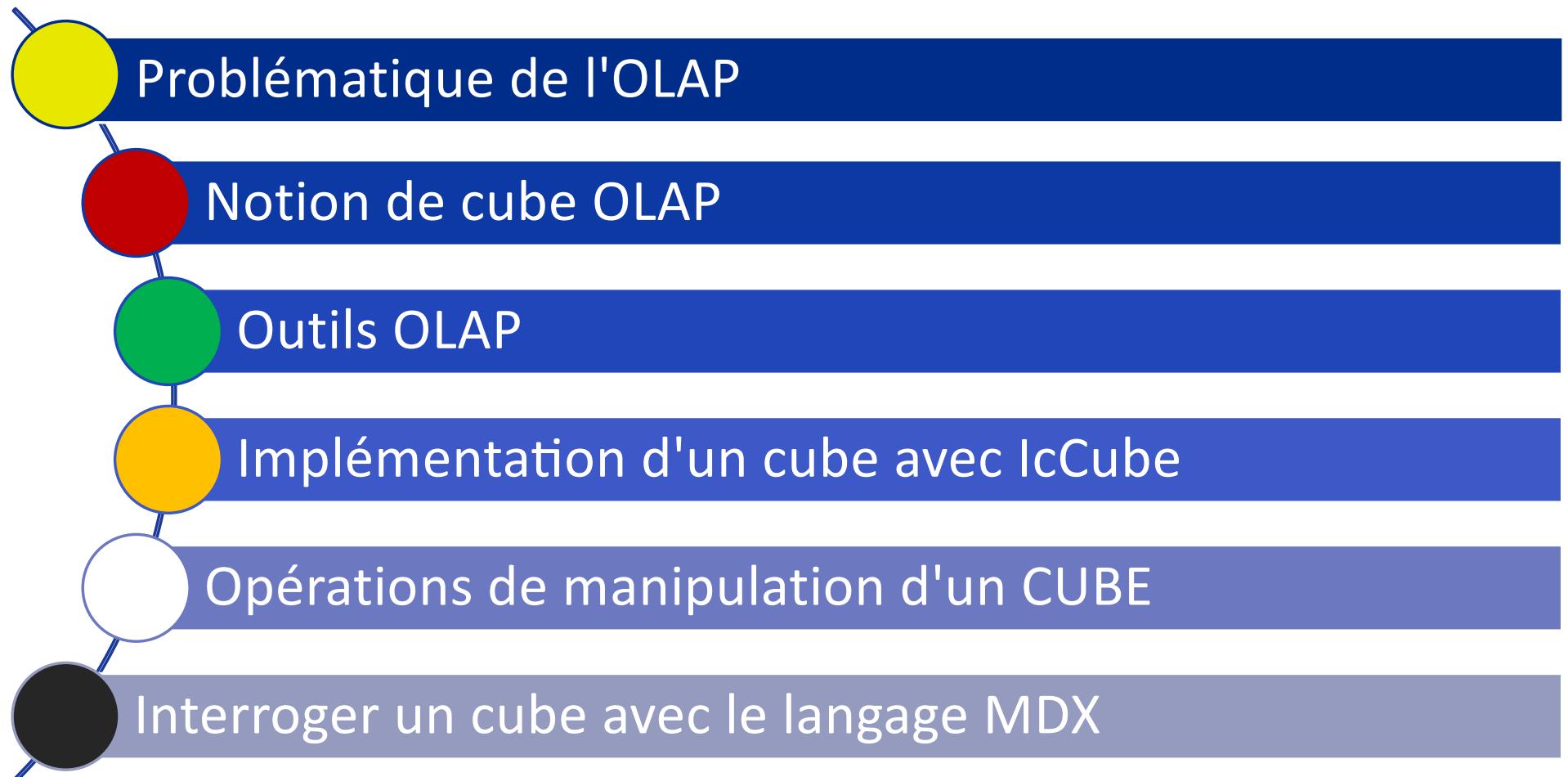


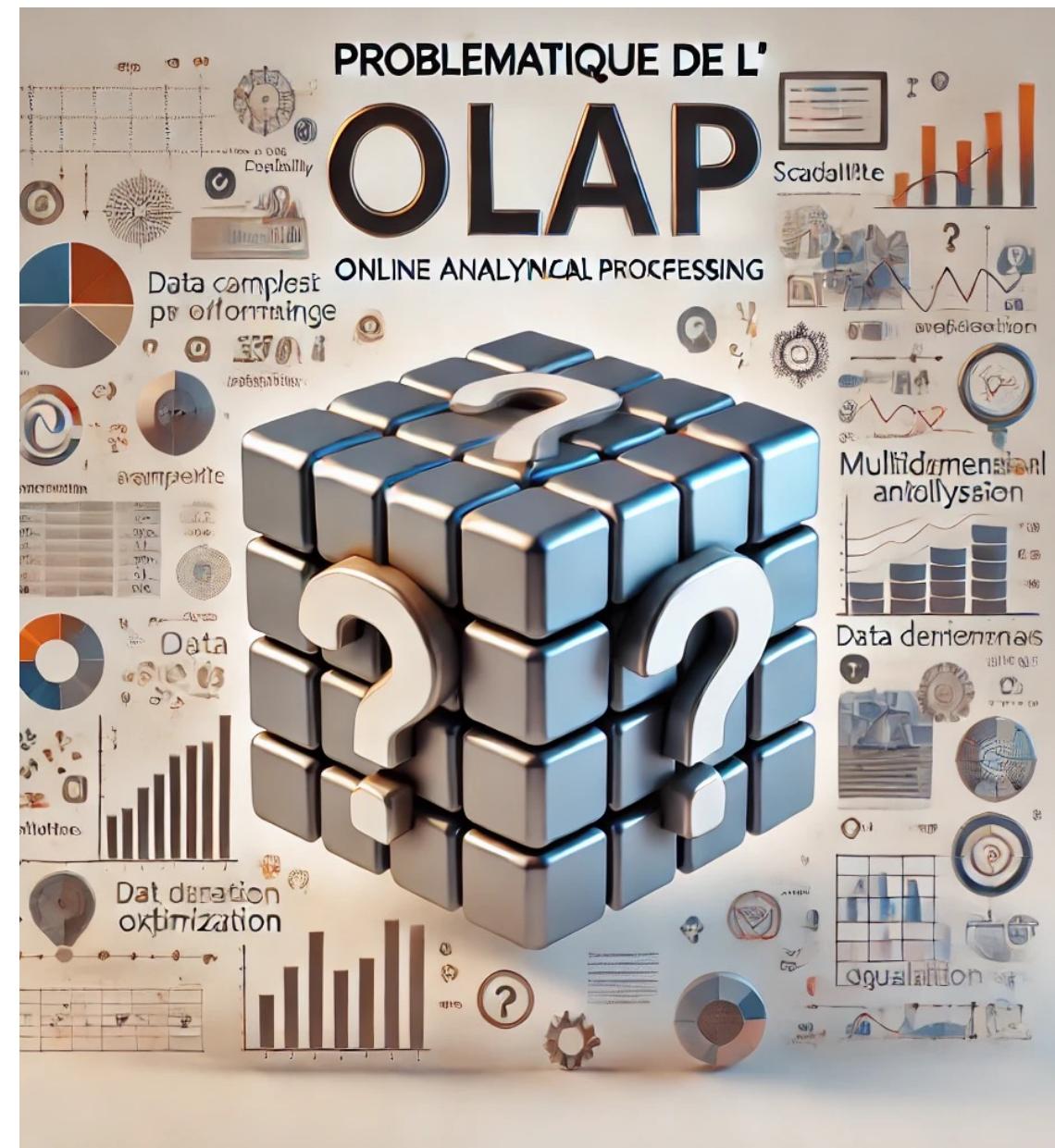
## CH4 – Exploitation d'un DW

- Principes OLAP
- Notion d'hypercube OLAP
- Langage MDX
- Mise en pratique avec IcCube



# CH4 – Exploitation d'un DW





# 1 – Problématique de l'OLAP

# Qu'est-ce que l'OLAP ?

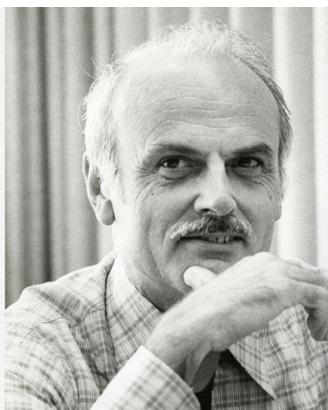
## OnLine Analytical processing

- Traitement Analytique en ligne
- Terme défini à l'origine par EF. CODD en 1993 à travers 12 règles
- Regroupe les concepts, technologies et outils permettant :
  - De modéliser des données dans des **cubes multidimensionnels** (concept d'hypercube OLAP)
  - D'exécuter des analyses complexes et interactives grâce à des **langages de requêtes spécifiques** (comme MDX)
  - De faciliter la prise de décision stratégique

# Les 12 règles de l'OLAP selon CODD

## Remembering Edgar F. Codd

The Father of Relational Database Management Systems



N°	Règle	Description
1	<b>Vue multidimensionnelle</b>	Organisation des données en dimensions et mesures dans des cubes OLAP.
2	<b>Transparence</b>	Fonctionne avec diverses sources de données sans complexité visible pour l'utilisateur.
3	<b>Accessibilité</b>	Interface utilisateur simple et compatibilité avec des langages standards (ex. MDX).
4	<b>Performances constantes</b>	Temps de réponse acceptable, quelle que soit la taille des données.
5	<b>Architecture client/serveur</b>	Répartition des charges entre le client et le serveur pour l'efficacité.
6	<b>Génération dynamique des agrégats</b>	Calcul des agrégats à la volée selon les besoins des requêtes.
7	<b>Manipulation multidimensionnelle</b>	Permet des interactions intuitives comme drill-down, roll-up, slice, dice.
8	<b>Absence de limite dimensionnelle</b>	Supporte un nombre illimité de dimensions et hiérarchies.
9	<b>Gestion des hiérarchies</b>	Support natif des hiérarchies pour naviguer entre niveaux de granularité.
10	<b>Analyse multi-utilisateur</b>	Requêtes simultanées par plusieurs utilisateurs.
11	<b>Sécurité des données</b>	Gestion fine des droits d'accès aux données.
12	<b>Flexibilité pour le reporting</b>	Génération de rapports personnalisables et interactifs.

## Sources

- Image : <https://www.linkedin.com/pulse/edgar-f-codd-father-relational-database-management-systems-pain-wszkc/>
- Article CODD : Salley, C., & Codd, E.F. (1998). Providing OLAP to User-Analysts: An IT Mandate. [http://www.estgv.ipv.pt/PaginasPessoais/jloureiro/ESI\\_AID2007\\_2008/fichas/codd.pdf](http://www.estgv.ipv.pt/PaginasPessoais/jloureiro/ESI_AID2007_2008/fichas/codd.pdf)

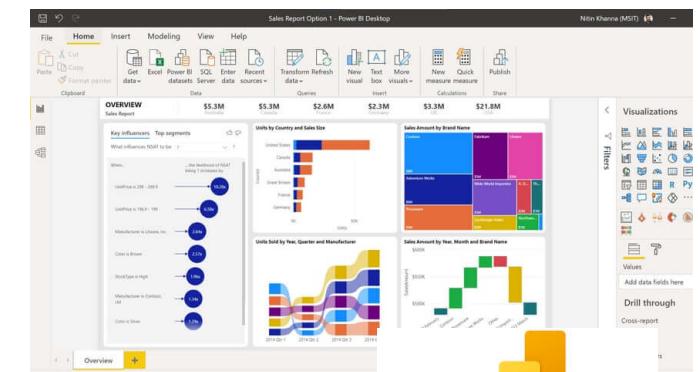
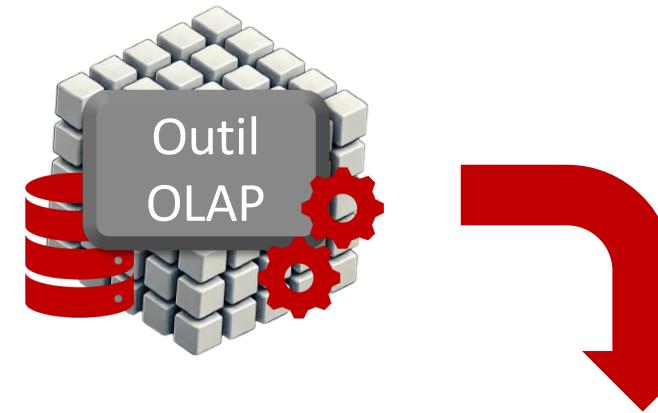
# Que regroupe l'OLAP ?

Concepts	Manipulation de données organisées en dimensions et mesures	<b>HyperCubes OLAP,</b> Hiérarchies, Modèles multidimensionnels
Outils	Plateformes permettant de définir, interroger et visualiser les cubes OLAP	<b>icCube</b> , Pentaho Mondrian, Microsoft SSAS, Tableau, Power BI
Langages de requêtes	Langages conçus pour interroger des données multidimensionnelles	<b>MDX (Multidimensional Expressions)</b> DAX (Data Analysis Expressions)

# OLAP versus DataViz

## Outils DataViz

- Focalisés sur la visualisation
- Ex: Tableau, Power BI, ou QlikView
- Objectifs
  - Créer des graphiques et tableaux de bord interactifs
  - Présenter visuellement des données pour les rendre compréhensibles aux utilisateurs métier
- Ces outils peuvent se connecter à des cubes OLAP pour visualiser les analyses réalisées dans les cubes



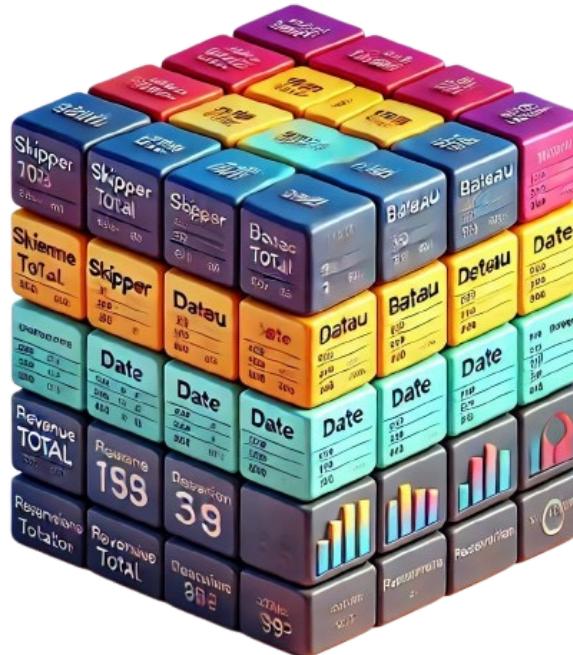
Power BI



## 2 – Notion d'hypercube OLAP

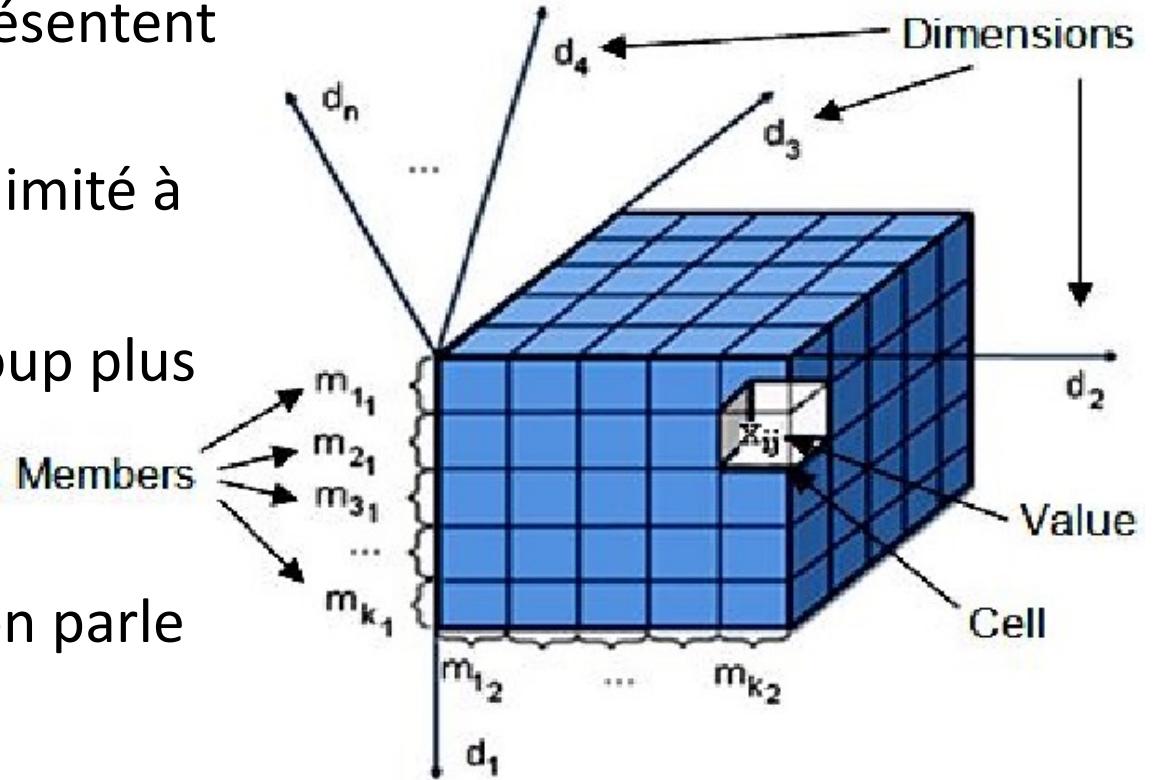
# Qu'est-ce qu'un hypercube OLAP?

- Un hypercube OLAP (ou cube OLAP) est une structure de données multidimensionnelle qui permet d'analyser rapidement des données selon plusieurs perspectives
- **Avantages**
  - **Performances élevées :**  
Calculs et agrégations rapides
  - **Analyse intuitive :**  
Permet de "couper" et "croiser" les données facilement



# Notion d'hypercube OLAP

- Même si les illustrations représentent souvent un cube :
  - Un "cube" OLAP n'est pas limité à 3 dimensions !
- Il peut avoir 4, 5 voire beaucoup plus de dimensions
- C'est la raison pour laquelle on parle d'hypercube



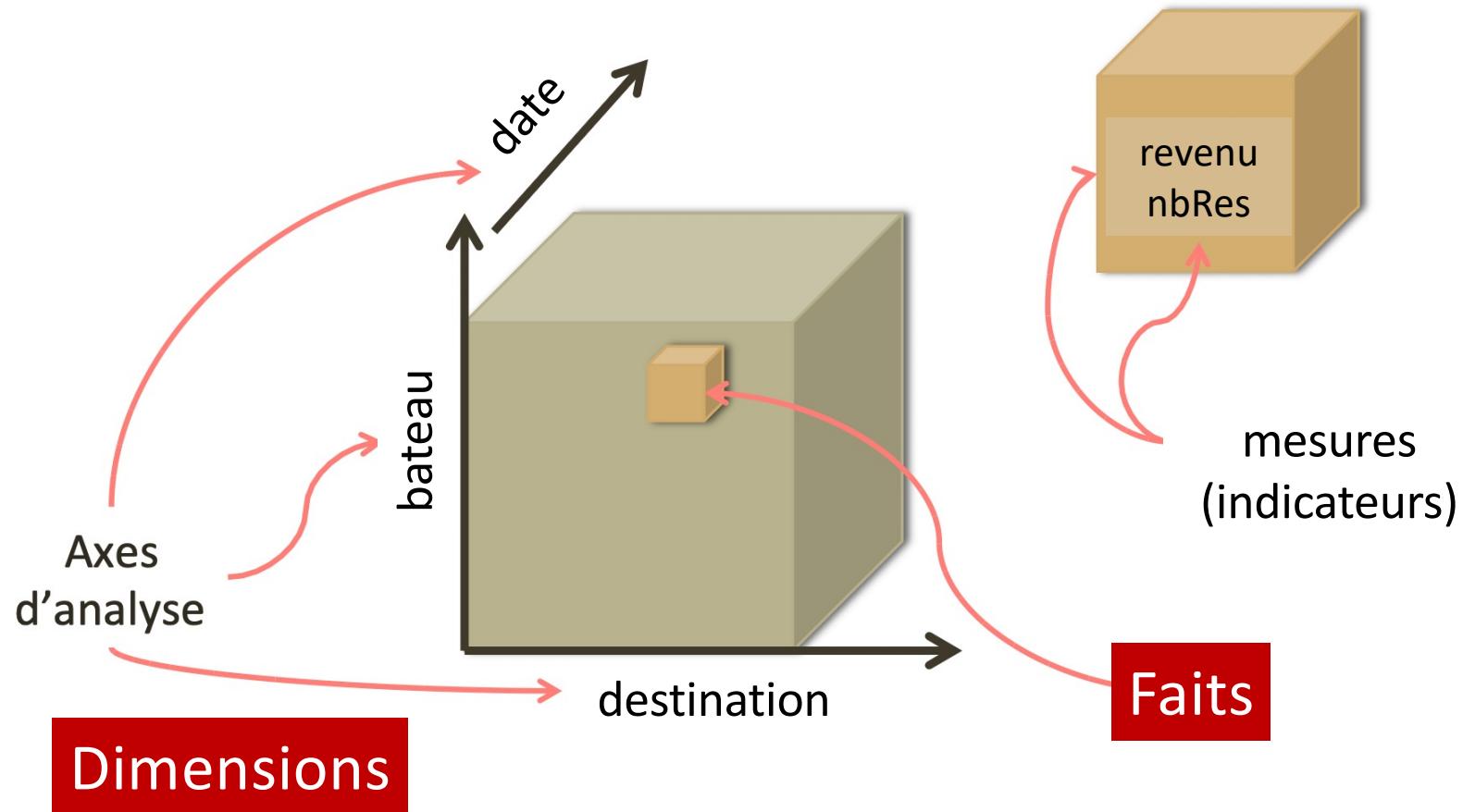
source image = [https://www.researchgate.net/figure/Multidimensional-OLAP-cube-The-sequence-theory-is-applied-for-the-dimensions-used-to\\_fig1\\_287066826](https://www.researchgate.net/figure/Multidimensional-OLAP-cube-The-sequence-theory-is-applied-for-the-dimensions-used-to_fig1_287066826)

# Composantes d'un cube OLAP

- Mesure ou indicateur = Contenu d'une Cellule
  - Attribut de la table des faits
  - Associé à une fonction d'agrégation : Sum, Count, ect...
- Dimensions = Axes d'analyse (ensemble de valeurs)
  - Associés à des tables de dimension
- Hierarchies
  - Spécifiées sur les dimensions afin de permettre des analyses à différents niveaux de détail
  - Niveaux (levels)
    - un ou plusieurs par hiérarchie
    - associé à une colonne de la table de dimension

Le choix des hiérarchies et des niveaux dépend des calculs souhaités

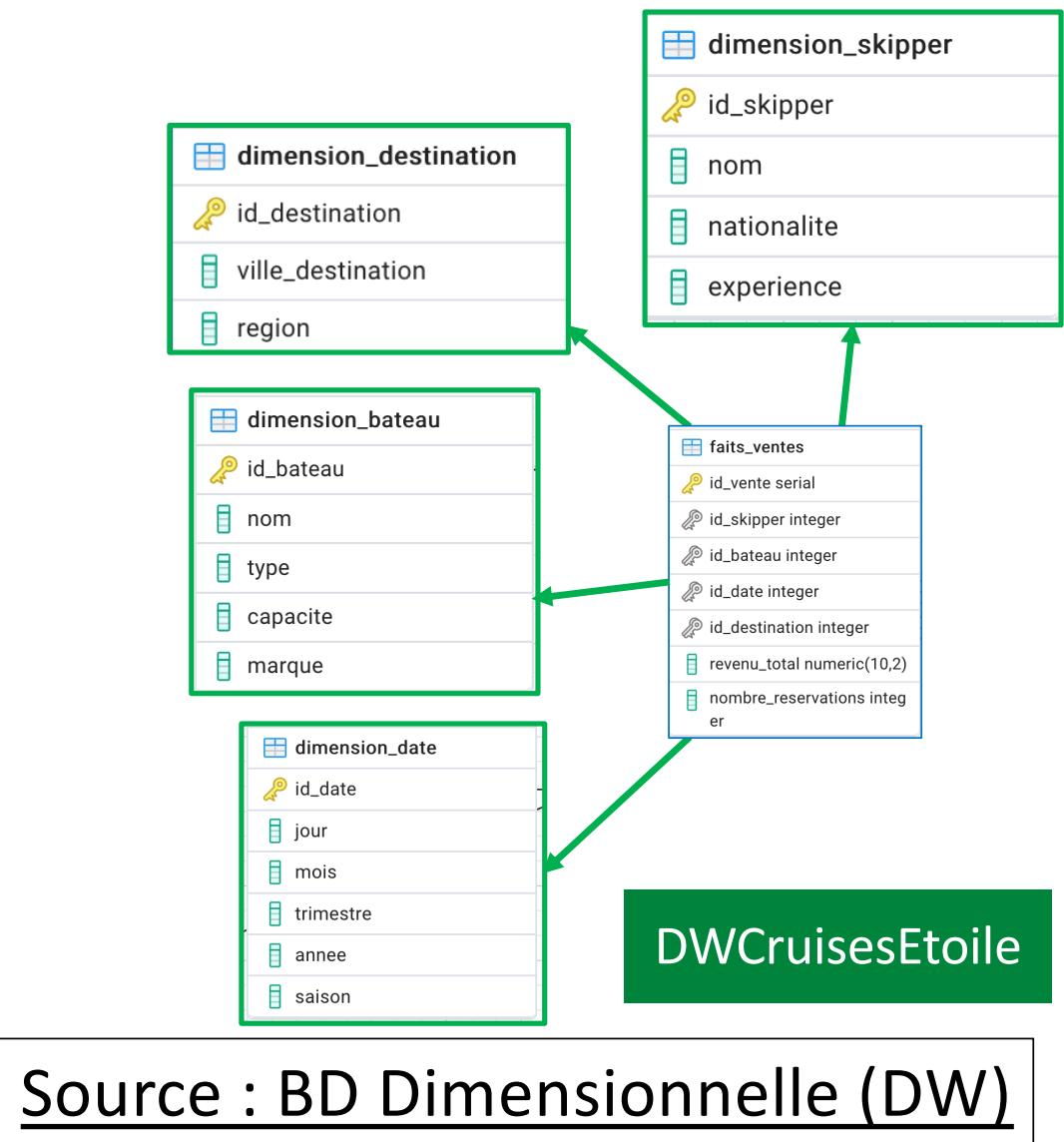
# Composantes d'un cube OLAP



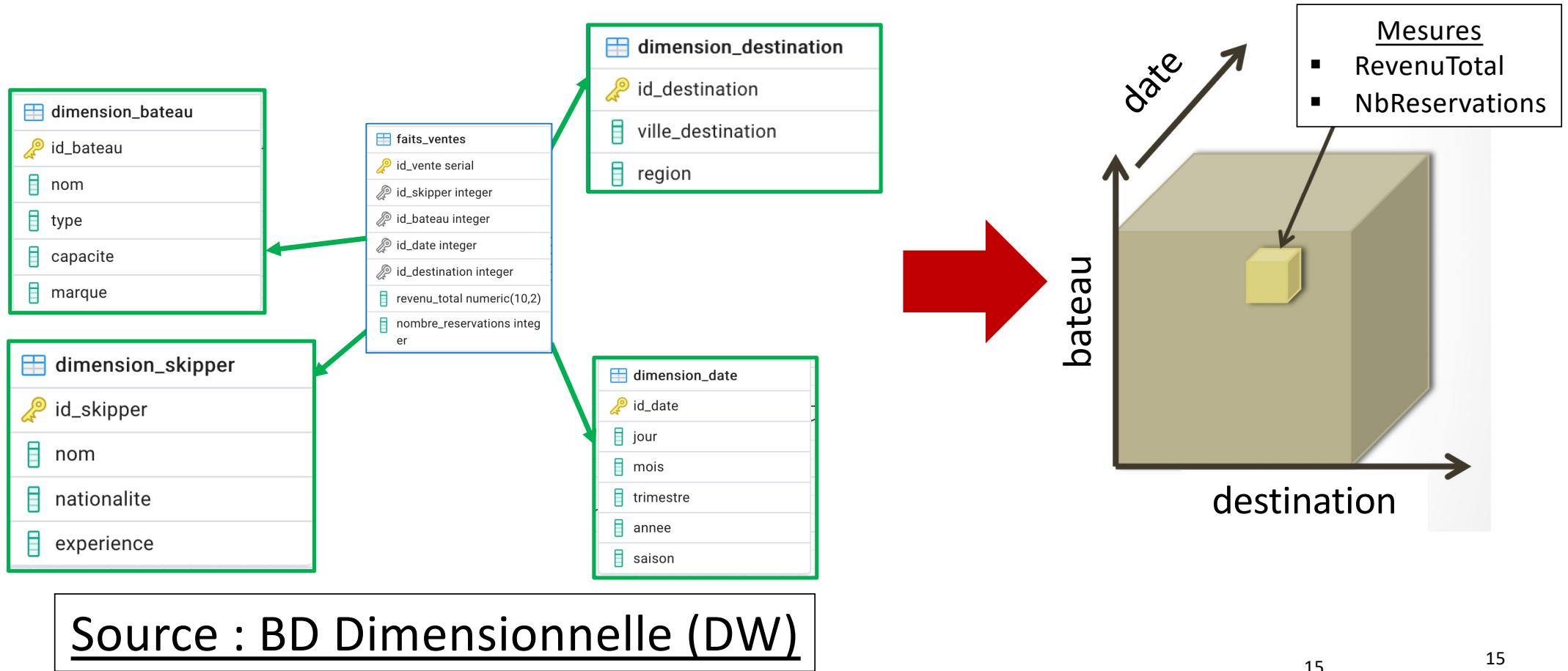
# Exemple de cube OLAP

## Objectifs d'analyse

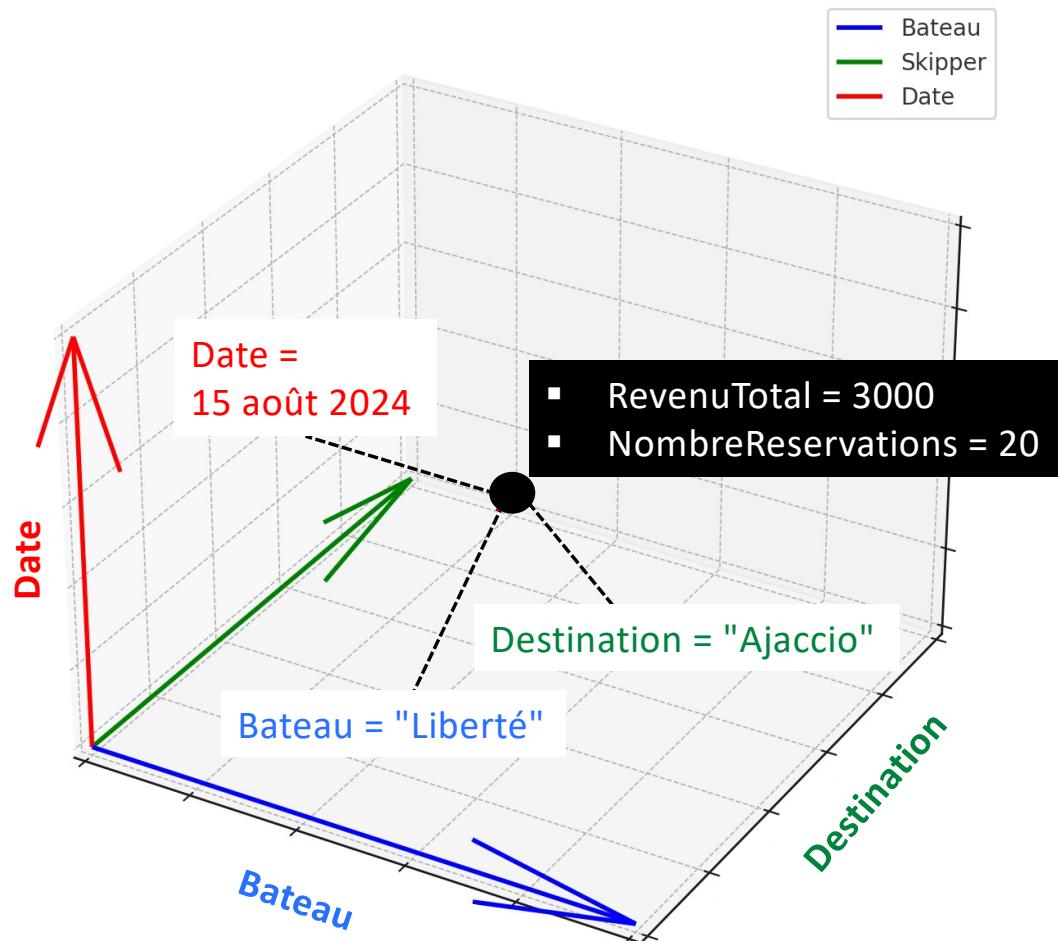
- Revenus totaux et nombre de réservations calculés par
  - nom de bateau
  - marque de bateaux
  - type de bateaux
  - région de destination et ville
  - année, mois, jour
  - mois



# Exemple de cube OLAP



# Exemple de cellule d'un cube OLAP

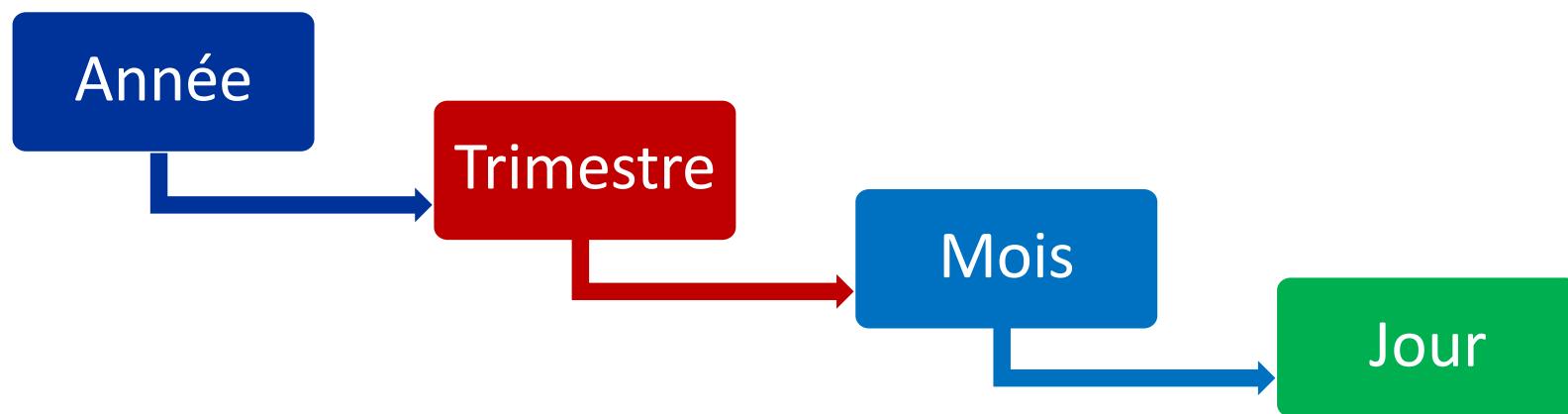


Ensuite, on pourra naviguer dans les dimensions pour obtenir rapidement des données agrégées :

- revenus totaux pour le bateau Liberté
- nombre de réservations pour le 15 aout 2024

# Notion de hiérarchie dans une dimension

- Définir une hiérarchie dans une dimension permet d'organiser les données à différents niveaux de granularité
- Exemple avec la dimension **Date**



# OLAP cube data model

Continuing introduction to OLAP cubes, it's important to mention their building parts – cube components. Let's start a bottom-up.

## Measure

- ▶ This is the real data in OLAP cubes. Measures are stored in their own dimensions, which are called measure dimensions.

## Member

- ▶ A single data item within dimension. The access to members is usually implemented through an OLAP structure of dimensions, hierarchies, and levels.

## Level

- ▶ Levels are categories that aggregate into a common hierarchy. You can perceive levels as data fields that can be queried and analyzed separately from each other.

## Hierarchy

- ▶ Predefined aggregation of levels in the specified dimension. OLAP cube hierarchy allows you to create summary data, such as computer programs for free, and analyze them at different levels of the structure, without delving into the relationship that exists between these levels.

## Dimension

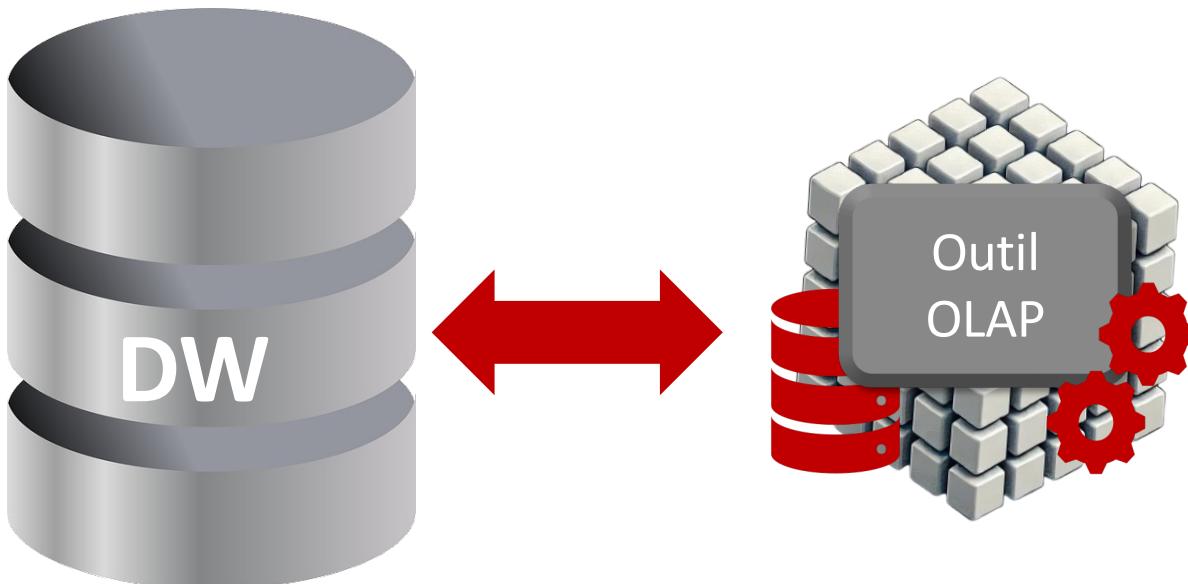
- ▶ The main characteristic of the analyzed data elements.

source : <https://galaktika-soft.com/blog/olap-cubes.html>

### 3 – Outils OLAP



# Quel outil pour créer un cube OLAP ?



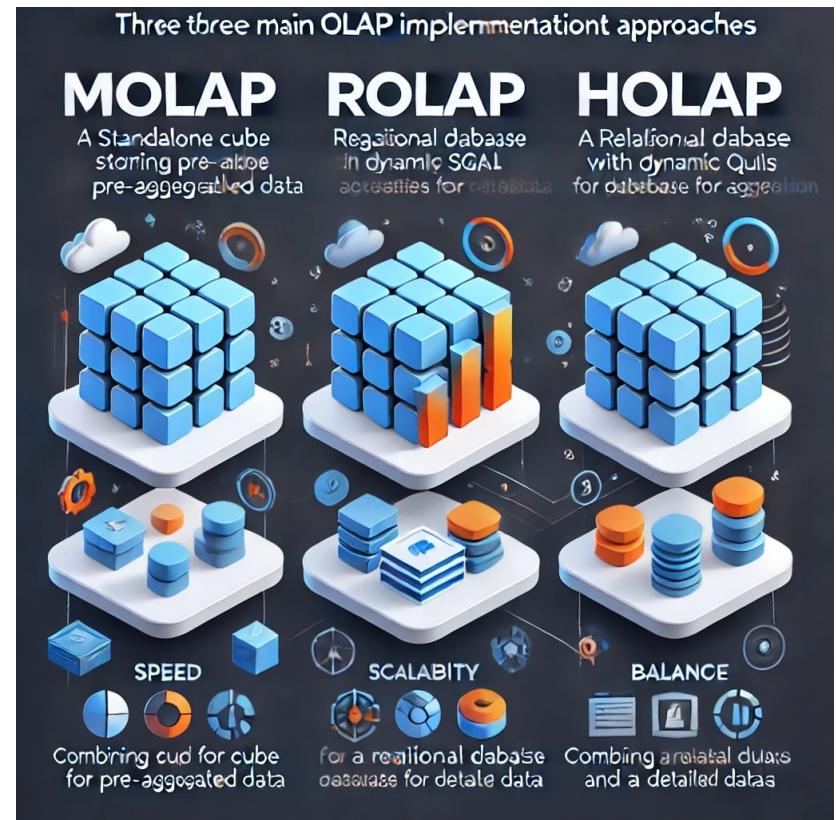
Un outil OLAP intègre  
des outils pour :

- Créer un cube OLAP
  - en se connectant à un DW pour extraire les données sources
- Stocker le cube
  - non systématique
- Requêter le cube
  - requêtes OLAP

# Catégories d'outils OLAP

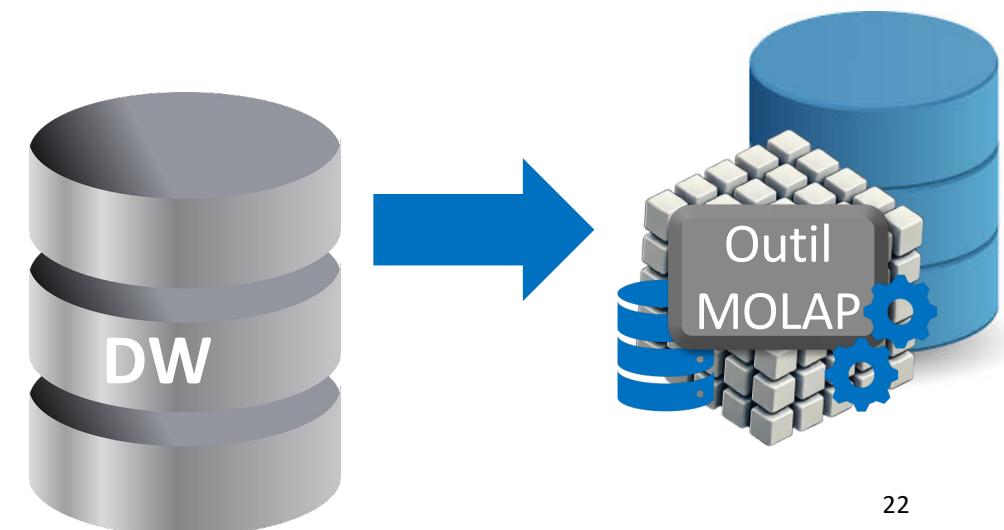
Trois principales approches d'implémentation :

- **MOLAP** : Multidimensional OLAP
  - Les Cubes sont stockés dans l'outil
- **ROLAP** : Relational OLAP
  - Les données restent dans le DW
- **HOLAP** : Hybrid OLAP
  - Cube + Connexion à une BDr



# MOLAP (*Multidimensional-OLAP*)

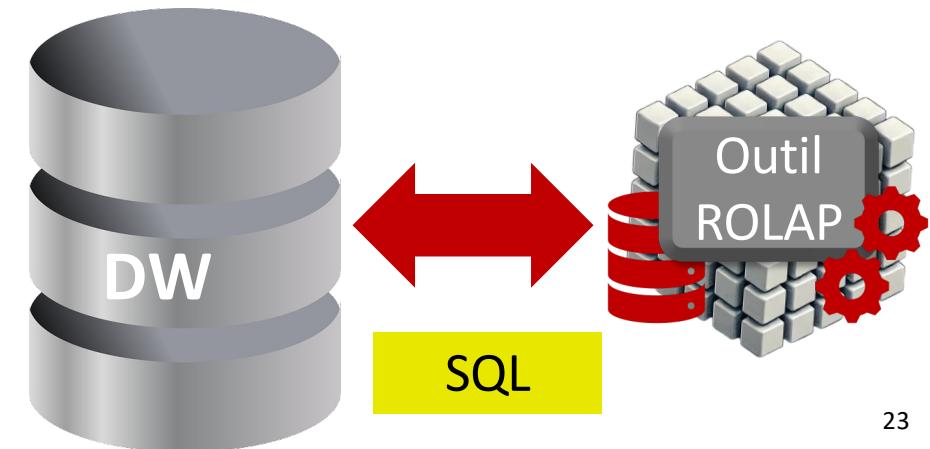
- Les données sont pré-calculées et stockées dans un format multidimensionnel spécifique (**cube OLAP physique**)
- Structure optimisée pour des requêtes **rapides**
- Limité par la taille du cube (ne convient pas à de très grandes bases de données)
- Outils
  - Microsoft Analysis Services (SSAS)
  - Oracle Essbase
  - IBM Cognos TM1



# ROLAP (*Relational-OLAP*)

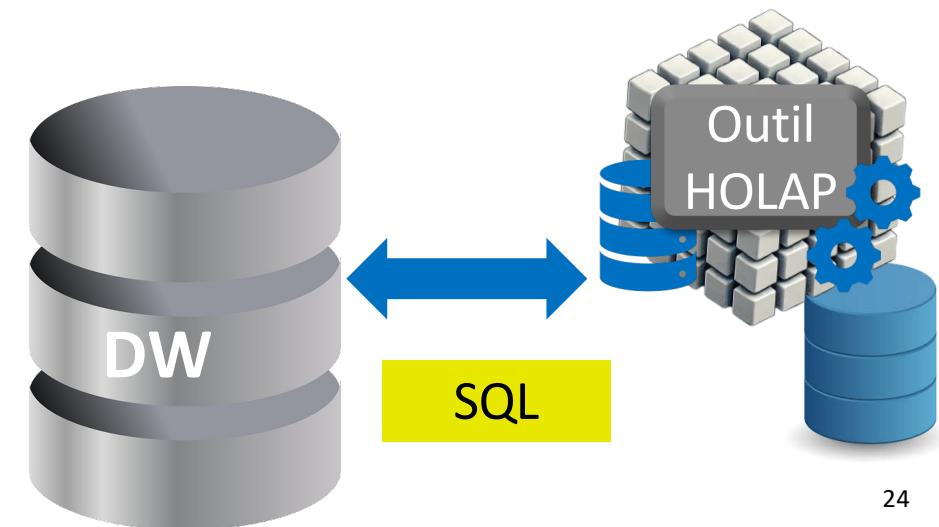
Approche la plus répandue

- Rien n'est stocké dans l'outil OLAP
  - Les données restent dans une base relationnelle (DW **SGBDR**)
  - Les requêtes OLAP sont traduites en **SQL**, et les agrégats sont calculés à la volée
- Requêtes plus lentes comparées à MOLAP (agrégats non pré-calculés)
- Peut gérer des bases de données massives
- Outils
  - **Pentaho Mondrian, ICCube**



# HOLAP (*Hybrid-OLAP*)

- Combine les avantages de **MOLAP** et **ROLAP** :
  - Les agrégats souvent utilisés sont stockés dans un format multidimensionnel (MOLAP).
- Les données détaillées restent dans le DW (BD relationnelle) (ROLAP)
- Outils
  - **Microsoft SSAS** : Mode hybride
  - **SAP BW**
  - **IBM Cognos Analytics**
  - **IcCube**

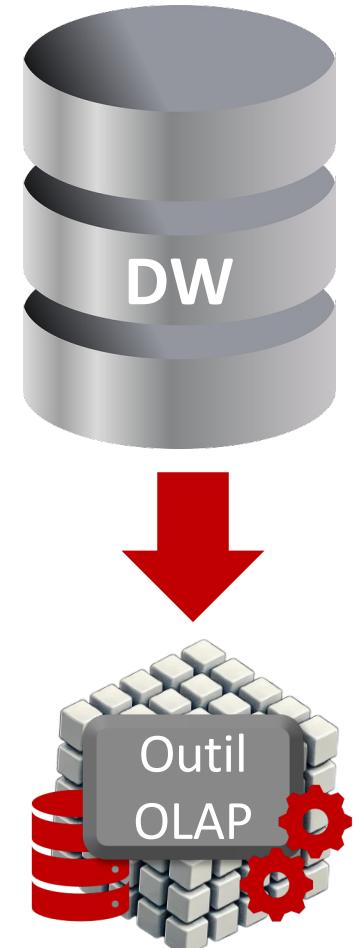


# Comparaison ROLAP, MOLAP, HOLAP

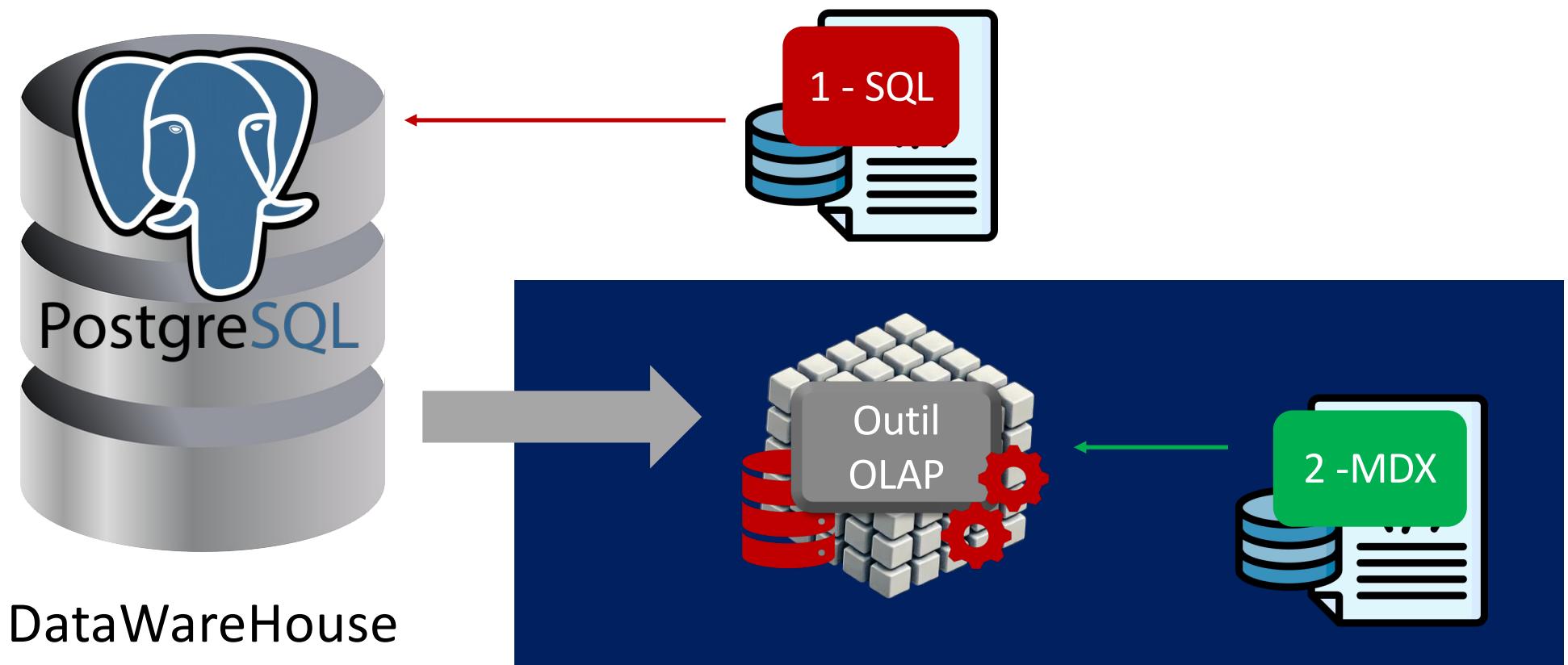
Critères	MOLAP	ROLAP	HOLAP
Stockage	Multidimensionnel	Relationnel	Hybride
Performance	Très rapide	Moyenne	Bonne
Scalabilité	Limitée	Très élevée	Élevée
Mise à jour	Lente	Rapide	Moyenne
Flexibilité	Faible	Très flexible	Flexible
Outils	Essbase, SSAS (multi)	Mondrian +Pentaho workbench , PostgreSQL	IcCube, SSAS (hybride), SAP BW

# Etapes de création d'un cube dans un outil OLAP

- Configuration de la connexion au Datawarehouse
- Définition du **schéma** du cube
  - Génération d'un fichier XML spécifique
- Deux possibilités d'implémentation :
  - Chargement des données du DW dans le cube (Mode HOLAP)
    - Stockage du cube dans l'outil
  - Les données restent dans le DW (mode ROLAP pur)
    - Le moteur OLAP requête le DW "à la volée" pour chaque requête sur le cube
- Requêtage du cube OLAP
  - Langage spécifique MDX



# Deux approches pour interroger des données multi-dimensionnelles



# Outil OLAP pour le cours

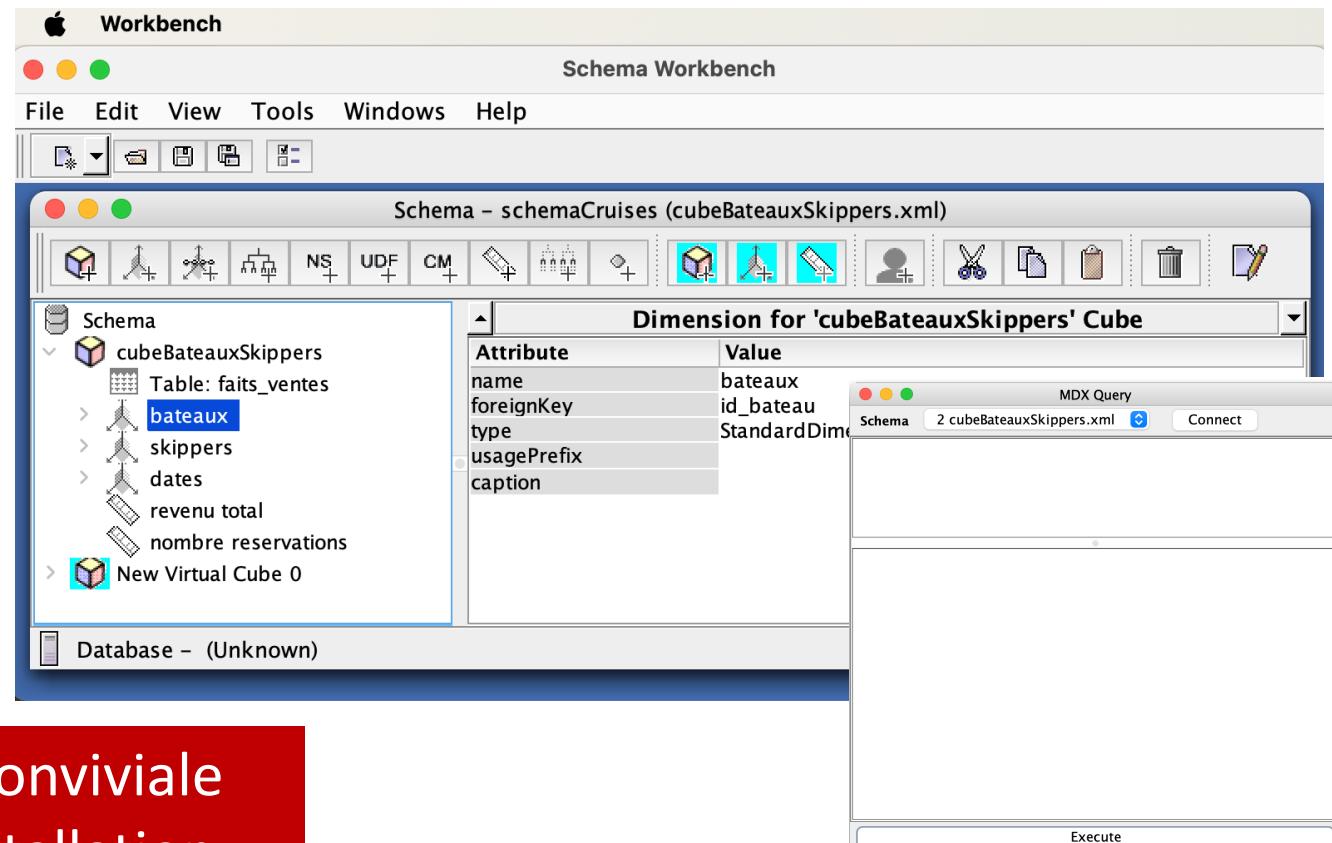
## Nos critères de choix

- Moteur OLAP
- Installation facile et rapide
- Connexion possible avec différentes sources :
  - ✓ BD postgGres
  - ✓ BD MongoDB
  - ✓ fichiers CSV
- Interface conviviale de création d'un cube OLAP
- Editeur du langage MDX pour le requêtage des cubes



# Schema workbench (+ Mondrian) Pentaho

- interface graphique + Mondrian (moteur OLAP)
- Version communautaire gratuite
- Composants de la suite Pentaho BI



Interface peu conviviale  
Difficultés d'installation

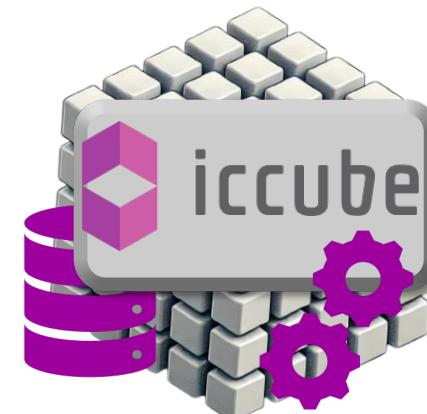
# Notre solution dans ce cours : IcCube

- Les données dimensionnelles sont stockées dans un DW (BD relationnelle PostgreSQL) (ROLAP)
- Un outil OLAP externe charge les données dans un cube multidimensionnel (MOLAP).



DataWareHouse

Chargement



stockage relationnel

stockage multidimensionnel en  
mémoire



**iccube**  
making data talk

## 4 - Implémentation d'un cube OLAP dans IcCube

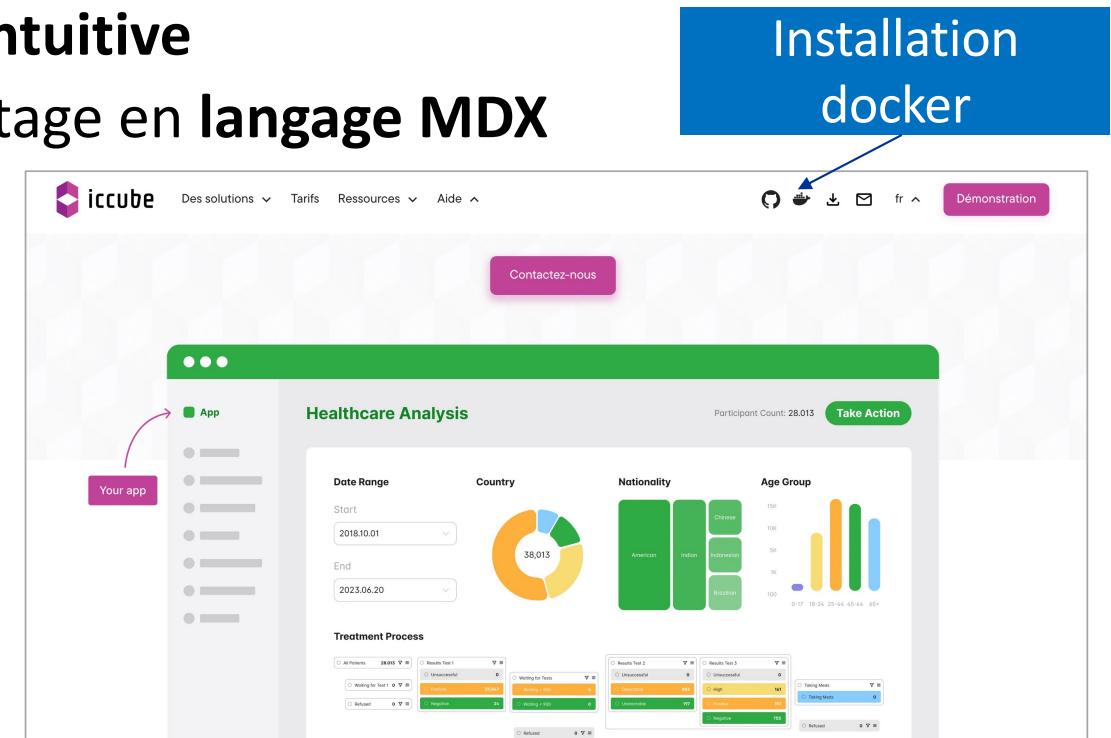
# iCCube



- Entreprise Suisse créée en 2010
- Solution BI pour l'analyse et l'exploration des données multidimensionnelles basée sur une **interface graphique intuitive**
  - Crédit à la création de **cubes OLAP** et requêtage en **langage MDX**
  - Outils interactifs de visualisation
- Déploiement local ou dans le cloud
  - Installation facile
- Support pour de nombreuses sources de données

logiciel propriétaire

Obtention d'une licence académique  
pour notre université



<https://www.iccube.com/fr>

# Installation icCube

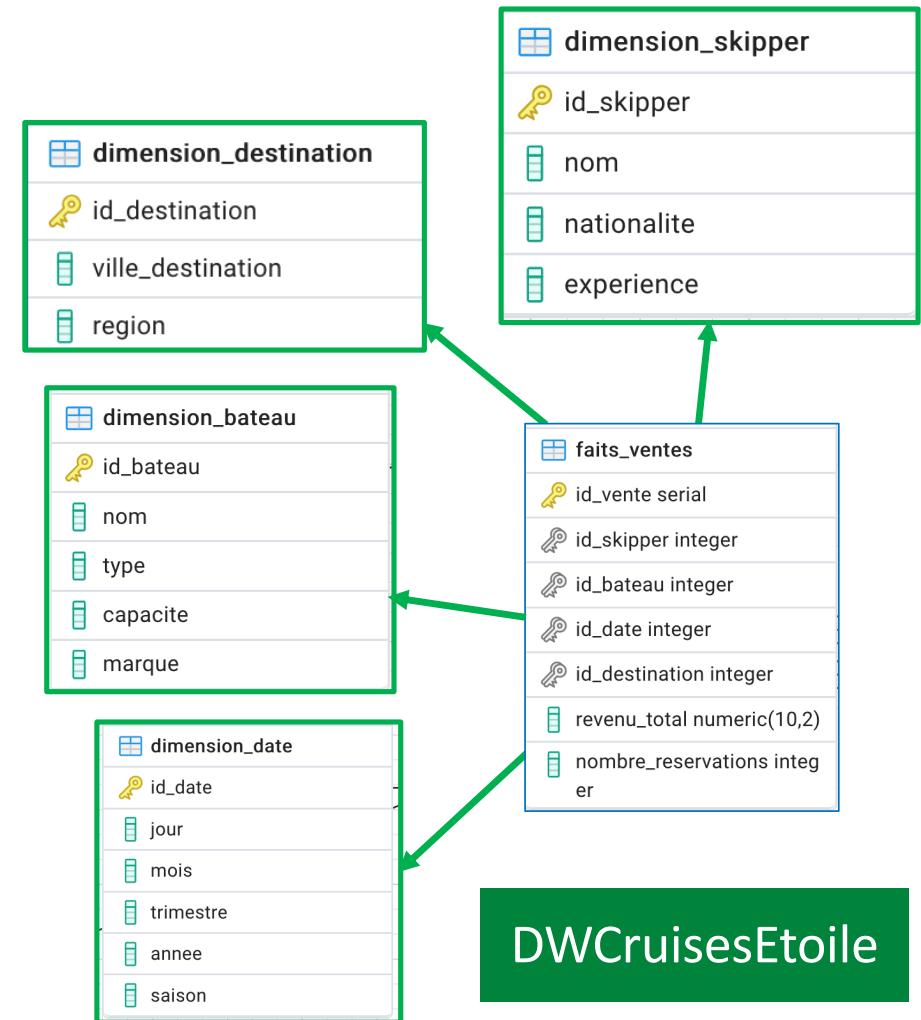
1. Récupérez sur l'ENT le fichier de licence ***icCube-4.lic***
  - il vous sera demandée lors du lancement de icCube après l'installation
2. Installez java21 ou une version ultérieure (si vous ne l'avez pas déjà)
3. Dans un terminal :
  - Téléchargez l'image docker associée  
`docker pull ic3software/iccube:8.4.14-chromium`
  - Lancez l'image  
`docker run -d --name icCube-eval --restart unless-stopped -p 8282:8282 \ic3software/iccube:8.4.14-chromium`
4. Pour faciliter les connexions entre plusieurs conteneurs, créez un réseau personnel docker (mon\_reseau par exemple) dans lequel seront placés vos conteneurs postgresql, pgadmin et icCube-eval
  - Vérifiez les noms de vos conteneurs (`docker ps`)
  - Ajoutez les conteneurs au réseau (`docker network connect mon_reseau serveurPostgres, ...`)
5. Configurez la source de donnée associée à votre BD postgresql avec les mêmes paramètres que dans pgadmin



# Exemple de cube OLAP

## Objectifs d'analyse

- Revenu total et nombre de réservations calculés par
  - nom de bateau
  - marque de bateaux
  - type de bateaux
  - région de destination et ville
  - année, mois, jour
  - mois



DWCruisesEtoile

Source : BD Dimensionnelle (DW)

# Etapes de création d'un cube avec IcCube

Créer un  
schéma

Créer un  
**cube**

Définir les  
**dimensions**  
-> **hierarchies**  
-> **levels**

Déployer et  
**charger** le  
cube

Définir les  
**sources**  
de  
données

Définir les  
**measures**

Définir **liens**  
**measures-**  
**dimensions**



# Lancement IcCube

The image shows a screenshot of the Iccube console builder interface. On the left, a modal window displays the Iccube logo and a login form with fields for 'User' (admin) and 'Password'. A 'Login' button is at the bottom. The main area features a header with the Iccube logo, 'Builder' tab, and a search bar. Below the header, the 'Overview' tab is selected, showing a list of schemas: 'DWcruises' and 'cubeSimple'. To the right, there's a sidebar with icons for 'Builder', 'Definition Cache', and 'Revisions', along with a help icon.

# Création d'un schéma

The screenshot illustrates the process of creating a new schema in the iccube Builder. The top part shows the main navigation bar with the iccube logo, 'Builder', a search bar, and user icons. A pink highlighted 'Overview' tab is selected. Below it, there are links for 'Schemas' and 'Definition Cache'. A large blue arrow points from a pink circle containing a white plus sign on the right side of the main panel towards a secondary, larger window below. This window is titled 'New Schema' and contains fields for 'Name \*' (filled with 'DWCruses'), 'Description' (filled with 'Analyse des ventes de croisières'), and 'Group \*' (filled with 'DBcours'). A 'SAVE' button is visible at the top right of this window.

iccube

Builder

search for ...

Overview

Schemas

Definition Cache

schema search

+

iccube

Builder

search for ...

Overview

Schemas

Definition Cache

Revisions

New Schema

SAVE

Name \*

DWCruses

Description

Analyse des ventes de croisières

Group \*

DBcours

# Connexion au DataWareHouse

The screenshot shows two views of the iccube Builder interface. On the left, the 'Home' view for the project 'DWcruises' is displayed. It features a sidebar with navigation links like Properties, Data Integration, Semantic Layer, Measures, Script, Links Overview, Structure Overview, Advanced, and Deployment. The main area contains four buttons: 'Create DataSource', 'Create DataTable', 'Create Measures', and 'Create Scripts'. A blue arrow points from the 'Create Measures' button towards the right-hand configuration screen. On the right, a detailed configuration screen for a 'New Data Source : Postgres' is shown. The screen has tabs for 'Type' (selected), 'Name', 'Properties', and 'Create'. Under the 'Type' tab, there are sections for 'SERVER', 'ADVANCED', and 'QUERY CONSOLE'. The 'SERVER' section contains fields for 'Server Name \*' (172.17.0.1) and 'Port Number' (5430). The 'ADVANCED' section contains a field for 'DB Name \*' (DWcruiseEtoile). The 'User \*' field (containing 'postgres') and the 'Password' field (containing '.....') are highlighted in yellow, indicating they are the focus of the comparison.

Connexion à votre DW sur  
votre serveur PostgreSQL  
(paramètres identiques à  
votre connexion dans  
pgAdmin)

# Création d'un cube

The screenshot shows the iccube Builder interface. On the left, a sidebar menu includes options like DWcruisesCours, Properties, Data Integration, Semantic Layer, Measures (which is selected and highlighted in pink), Script, Links Overview, Structure Overview, Advanced, and Deployment. The main workspace is titled "New Cube" and contains fields for "Name \*" (with "CubeCruises" typed in) and "Description". A "SAVE" button is at the top right. A large blue callout box on the right states "Un schéma peut contenir plusieurs cubes". A pink callout bubble points from the "MEASURE GROUP" button on the right towards the "Name" field.

iccube

Builder

DWcruisesCours

New Cube

Properties

filter

Data Integration

Semantic Layer

Measures

Script

Links Overview

Structure Overview

Advanced

Deployment

search for ...

SAVE

Name \*

CubeCruises

Description

+ MEASURE GROUP

Un schéma peut contenir plusieurs cubes

# Création des mesures (measures)

**Left Panel:** DWcruisesCours - New Measure Group

**Top Step:** Setup Type (NEXT button highlighted)

**Bottom Step:** Measures (NEXT button highlighted)

**Right Panel:** Measure Group : Ventes

Detailed description: This diagram illustrates the process of creating a measure group named "Ventes". It shows two screenshots of a software interface. The top screenshot shows the initial setup step where the user selects a data table ("public.faits\_ventes") and names the measure group ("Ventes"). The bottom screenshot shows the "Measures" step where the user defines various measures, such as "RevenuTotal" and "NbReservations", along with their corresponding columns and aggregation types. A dashed arrow points from the "NEXT" button in the top step to the "Measures" section in the bottom step. To the right, a summary of the created measure group is shown, including its name ("Ventes"), cube ("CubeCruises"), and two measures: "RevenuTotal" and "NbReservations".

# Création des dimensions->hiérarchie(s)->levels

- Une **dimension** est composée d'une ou plusieurs **hiérarchies**
- Une **hiérarchie** est caractérisée par des niveaux (**levels**)
- La position des levels est importante :
  - Elle conditionne les agrégations
  - La clé primaire de la table dimension associée dans le DW doit être présente dans chaque hiérarchie (en dernier)

+ LEVEL



	dimension_destination
🔑	id_destination
⌚	ville_destination
⌚	region

table dans le DW

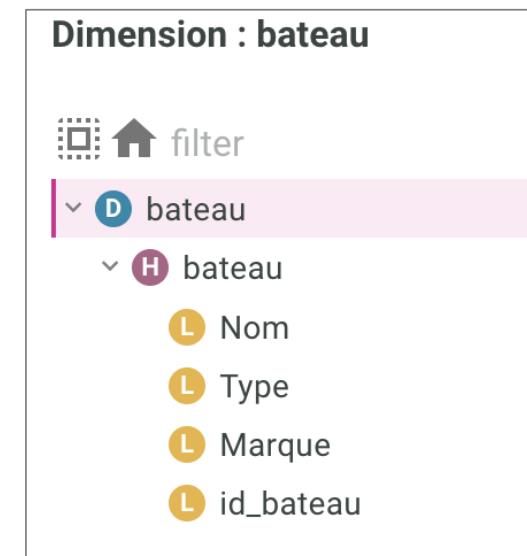
# Comment choisir les hiérarchies et les niveaux ?

On veut pouvoir faire des calculs

- pour chaque nom de bateau
- avec éventuellement des détails sur le type et la marque du bateau

On ne pourra pas avoir de calculs globaux pour une marque ou pour un type indépendamment du nom du bateau

Choix 1



# Comment choisir les hiérarchies et les niveaux ?

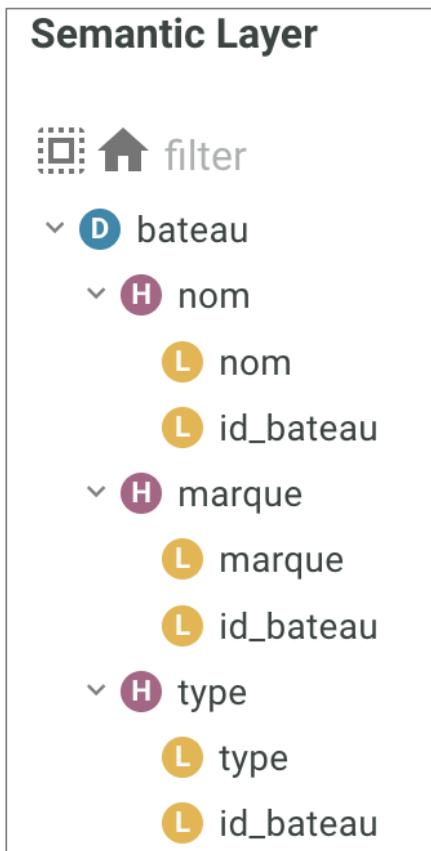
On veut pouvoir faire des calculs

- pour chaque nom de bateau
- pour chaque **marque** indépendamment du type et du nom
- pour chaque **type** indépendamment du nom

	RevenuTotal
▼ All	86,267.53
➤ Catamaran	4,867.02
➤ Yacht	81,400.51

Choix 2

On ne pourra pas avoir de détails pour un nom (sa marque et son type)

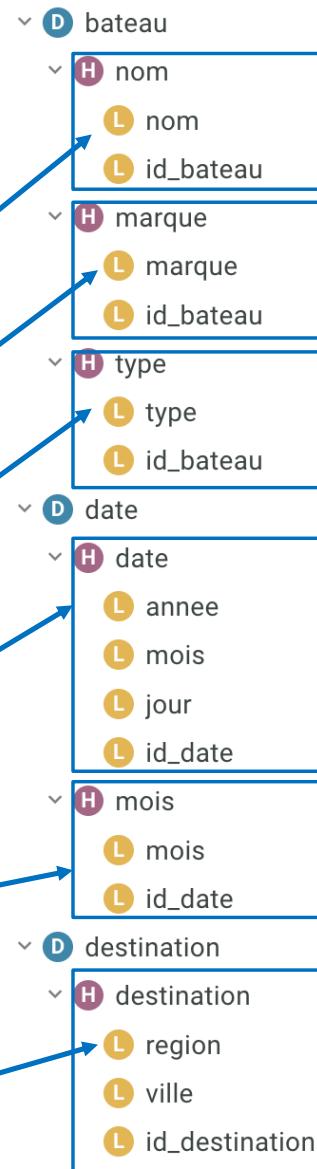


# Cube OLAP de l'exemple

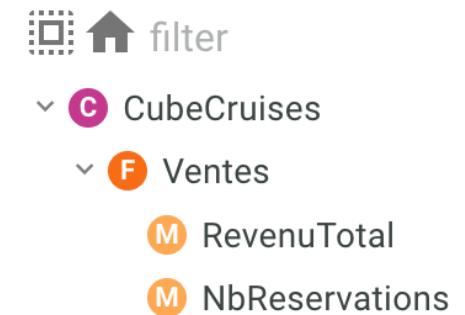
## Objectifs d'analyse

- Revenu total et nombre de réservations calculés par

- nom de bateau
- marque de bateau
- type de bateau
- année, mois, jour
- mois
- région de destination et ville



## Measures



**CubeCruises**

# Création des dimensions -> hiérarchie(s) ->levels

The diagram illustrates the process of creating a dimension in the iccube Builder interface, specifically focusing on the Semantic Layer.

**Initial Step:** The first screenshot shows the "Semantic Layer" section of the Builder. A blue callout box labeled "Create Dimension" is present, with the sub-instruction: "Add a new dimension definition into the semantic layer."

**Intermediate Step:** The second screenshot shows the "New Dimension" screen. A blue arrow points from the "Type" field in this screen to the "Type" field in the "Multi Levels Wizard" screen.

**Final Step:** The third screenshot shows the "LEVEL" configuration screen for the dimension. It includes fields for "NAME" (set to "bateau"), "PROPERTIES" (selected), "CONSTRAINTS", and "MEMBER PR". The "PROPERTIES" tab is active, showing settings like "Key Columns" (set to "id\_bateau<Integer>"), "Name Column" (set to "nom<String>"), and "Order Members" (set to "None").

**Annotations:**

- A blue arrow points from the "Type" field in the "New Dimension" screen to the "Type" field in the "Multi Levels Wizard" screen.
- A blue arrow points from the "Multi Levels Wizard" screen to the "LEVEL" configuration screen.
- A pink button labeled "+ HIERARCHY" is located above the "Multi Levels Wizard" screen.
- A pink button labeled "+ LEVEL" is located above the "LEVEL" configuration screen.
- The "NAME" field in the "LEVEL" configuration screen is highlighted in pink.
- The "SAVE" button is located in the top right corner of the "LEVEL" configuration screen.
- The page number "45" is located in the bottom right corner of the "LEVEL" configuration screen.

# Définition des liens mesures-dimensions

Measure Group : Ventes

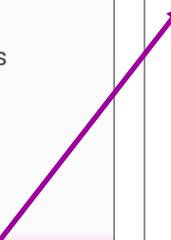
SAVE

NAME	LINKS	MEASURES	ADVANCED
filter C CubeDateBateauxSkip... <b>F Ventes</b> M Revenu Total M Nombre Reservati...	Dimension	FROM Level(s)	TO Facts
	bateau	Last Level (default) ▾	id_bateau id_bateau<Integer>
	date	Last Level (default) ▾	id_date id_date<Integer>
	destination	Last Level (default) ▾	id_destination id_destination<Integer>
			+ MEASURE

# Fichier xml du cube (vision partielle: dimension bateau)

DWcruises
Properties
▶ Data Integration
Semantic Layer
Measures
Script
Links Overview
Structure Overview
▼ Advanced
Perspectives
Localization
Colors
XML Definition
Deployment

```
<multiLevelDimension dynHierAllMember="All $HierarchyName" includeUsedColumn="true" dataTableId="d229d57b-8c08-469d-83a8-a7eeb70fc883" isDefaultTimeDimension="false" isIndexingByRange="false" isTimeDimension="false" id="ddde4353-0bdc-4aa8-8b1e-835949c0de9f" name="bateau">
  <multiLevelHierarchy hasAllLevel="true" allLevelName="All-Level" allMemberName="All" name="nom" isDefault="true" uuid="3d92d391-3b40-4170-9022-fd41f7635c56">
    <factAggregationType>MEMBER AND ANCESTORS</factAggregationType>
    <level uuid="ecc91884-2b14-4109-9d5d-c4aaa230d471" name="nom" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <nameCol name="nom"/>
      <orderType>NONE</orderType>
    </level>
    <level uuid="8c05d384-f1b0-42df-9f63-c41d2c0462ab" name="id_bateau" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <column name="id_bateau"/>
      <nameCol name="id_bateau"/>
      <orderType>NONE</orderType>
    </level>
  </multiLevelHierarchy>
  <multiLevelHierarchy hasAllLevel="true" allLevelName="All-Level" allMemberName="All" name="marque" isDefault="false" uuid="a3dc134e-b8ba-41bd-9c75-14bb7ba70f6f">
    <factAggregationType>MEMBER AND ANCESTORS</factAggregationType>
    <level uuid="974c0ec2-2721-4183-a488-037dbb4623ff" name="marque" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <nameCol name="marque"/>
      <orderType>NONE</orderType>
    </level>
    <level uuid="daad72ce-a4ec-473-b603-0e6f2c01d4ce" name="id_bateau" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <column name="id_bateau"/>
      <nameCol name="id_bateau"/>
      <orderType>NONE</orderType>
    </level>
  </multiLevelHierarchy>
  <multiLevelHierarchy hasAllLevel="true" allLevelName="All-Level" allMemberName="All" name="type" isDefault="false" uuid="4ada284d-c772-4e49-bc27-99025a0413a8">
    <factAggregationType>MEMBER AND ANCESTORS</factAggregationType>
    <level uuid="c4b418a9-02e1-4360-b6c1-4bb08e44b031" name="type" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <nameCol name="type"/>
      <orderType>NONE</orderType>
    </level>
    <level uuid="492bcdb9-d330-44e7-aa35-8f15f22bdf18" name="id_bateau" ignoreNameCollision="false" nameUnique="true" nameUniqueInParent="false" keyUnique="true">
      <column name="id_bateau"/>
      <nameCol name="id_bateau"/>
      <orderType>NONE</orderType>
    </level>
  </multiLevelHierarchy>
```



# Déploiement et chargement du cube

The screenshot shows the Cubicenter interface for the 'DWcruises' schema. On the left, a sidebar lists navigation options: Properties, Data Integration, Semantic Layer, Measures, Script, Links Overview, Structure Overview, Advanced, and Deployment. The 'Deployment' option is highlighted with a pink background. The main area is titled 'Deployment' and contains three buttons: 'Validate' (Check the validity of the schema definition), 'Deploy' (Copy the schema file to the runtime directory for later loading), and 'Deploy & Load' (Deploy the schema definition and load the schema). A purple arrow points from the 'Deploy & Load' button to a modal window. The modal window has a title 'Full Reload' and a message 'The schema 'DWcruises' has been successfully loaded and is accessible via the Admin application.' It features three buttons at the bottom: 'MDX', 'ADMIN', and 'OK'. A yellow callout at the bottom left states 'Le cube est calculé et chargé en mémoire, prêt à être requêté en MDX !'.

DWcruises

Properties

Data Integration

Semantic Layer

Measures

Script

Links Overview

Structure Overview

Advanced

Deployment

Deployment

Validate

Check the validity of the schema definition.

Deploy

Copy the schema file to the runtime directory for later loading.

Deploy & Load

Deploy the schema definition and load the schema.

**Full Reload**

The schema 'DWcruises' has been successfully loaded and is accessible via the Admin application.

MDX

ADMIN

OK

Le cube est calculé et chargé en mémoire,  
prêt à être requêté en MDX !

# A vous de jouer (1) ..



- Récupérez sur l'ENT le fichier DWcruises.icc-schema
- Téléchargez le schéma dans icCube

chemin vers le fichier  
.icc-schema

The screenshot shows the icCube web interface. On the left, the 'Schemas' tab is selected in the sidebar. The main area displays a table of schemas with columns for Name, Group, Definition File Updated Time, and File Name. Two rows are visible: 'DWcruisesExercice' (Group: BDCours, Updated: 26 Nov 2024 10:20:11 UTC, File: DWcruisesExercice.icc-schema) and 'DWcruises' (Group: DBcours, Updated: 24 Nov 2024 23:55:37 UTC, File: DWcruises.icc-schema). A red arrow points from the text 'chemin vers le fichier .icc-schema' to the 'Upload Schema Definition' dialog box. This dialog has a central area for file upload with the placeholder 'click/drop files here to upload (\*.icc-schema)' and a 'DONE' button at the bottom right.

Name	Group	Definition File Updated Time	File Name
DWcruisesExercice	BDCours	26 Nov 2024 10:20:11 UTC	DWcruisesExercice.icc-schema
DWcruises	DBcours	24 Nov 2024 23:55:37 UTC	DWcruises.icc-schema

## A vous de jouer (2) – Modification d'un cube..

- Modifiez la connexion au datawarehouse DWcruises pour que cela corresponde avec la configuration de votre serveur posgreSQL et de votre DW
- Ajoutez une dimension *skipper* dans le cube *CubeCruises*
- Définissez les hiérarchies nécessaires aux calculs suivants :
  - revenuTotal et NbReservations par noms de skipper avec éventuellement détails sur leur nationalité et leur expérience
  - revenuTotal et NbReservations par nationalité de skipper indépendamment de leur nom et leur expérience



Prenez soin de définir le lien des mesures avec votre nouvelle dimension

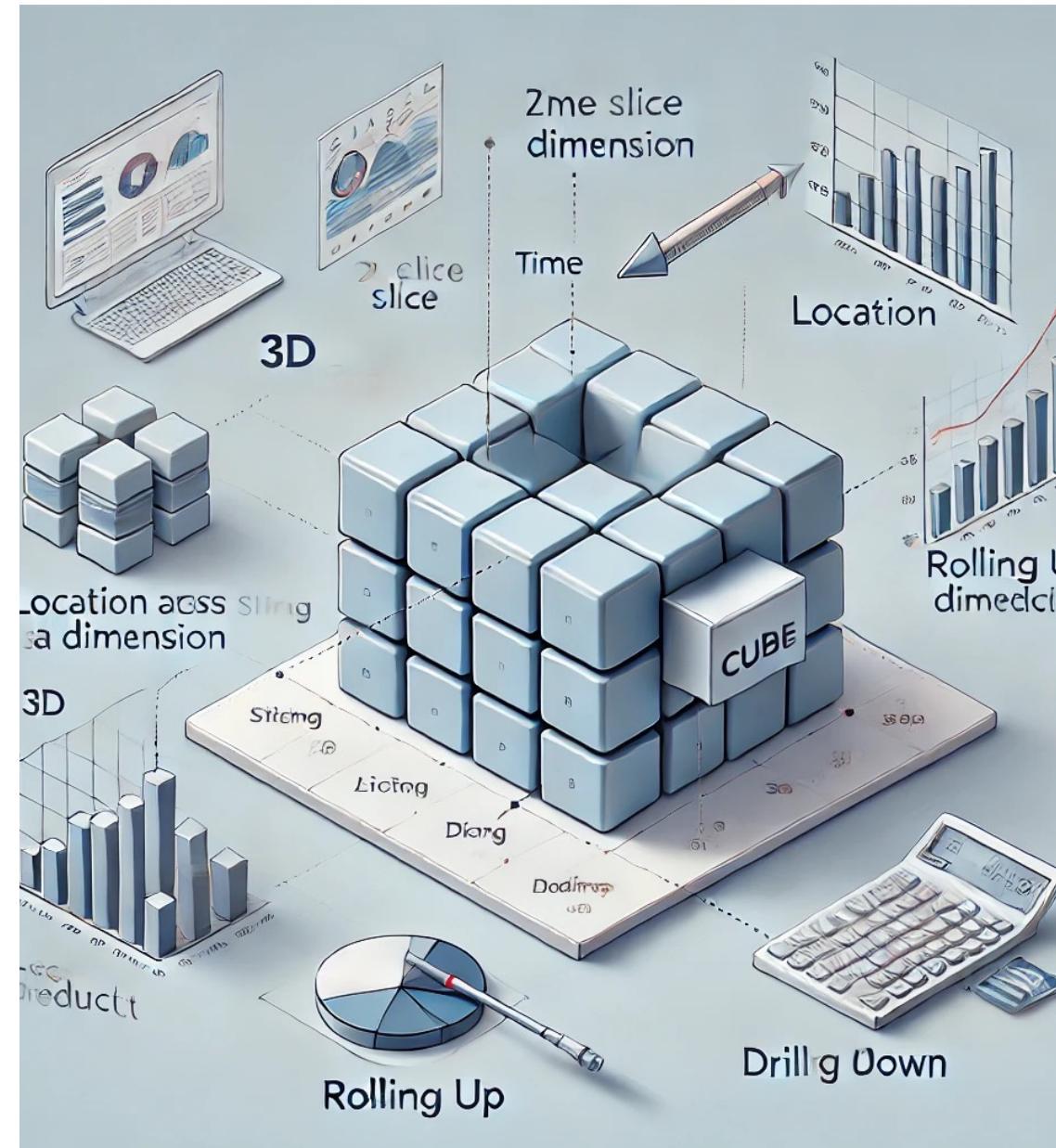
# A vous de jouer (3) – Définition de votre cube personnel ...

- Définissez un cube OLAP sous icCube alimenté par votre datawarehouse DWcruisesPerf (cf. CH3)

Questions auxquelles devra répondre l'analyse du cube :

1. Quelle est la durée moyenne pour l'ensemble des croisières ?
2. Combien de passagers ont été transportés par chaque bateau ?
3. Quels sont les deux bateaux qui ont transporté le plus de passagers?
4. Quelle a été la durée moyenne des croisières pour chaque bateau ?
5. Quelle est la durée moyenne des croisières gérées par chaque skipper ?
6. Quel est le nombre total de passagers embarqués par chaque skipper ?
7. Quel est le nombre total de passagers embarqués par niveau d'expérience des skippers ?
  - $0 < \text{expérience} < 10$  : débutant -  $10 \leq \text{expérience} < 20$  : intermédiaire -  $20 \leq \text{expérience}$  : expert
8. Quelle est le nombre de passager par skipper pour chaque bateau?

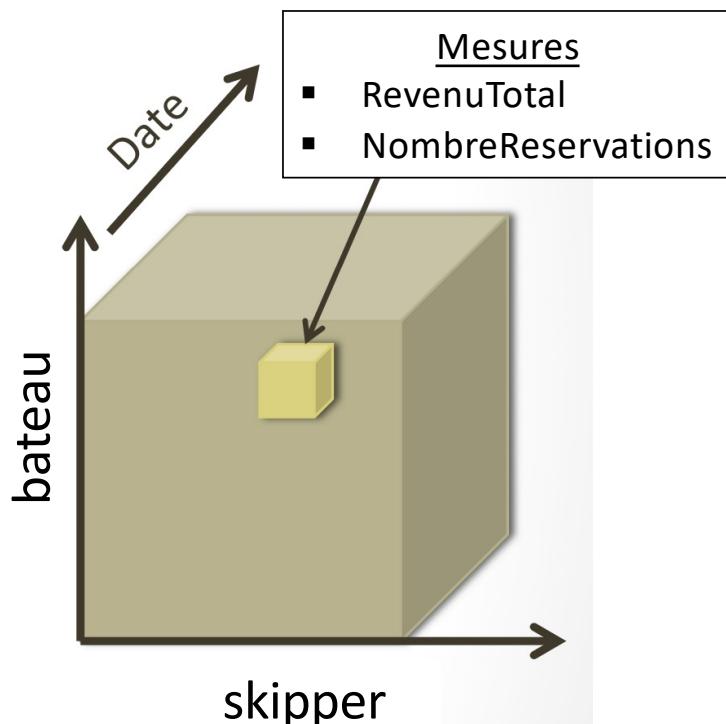




## 5 – Opérations sur des cubes OLAP

# Notion d'opération OLAP

- Les opérations OLAP offrent des outils puissants pour naviguer dans les données d'un cube OLAP

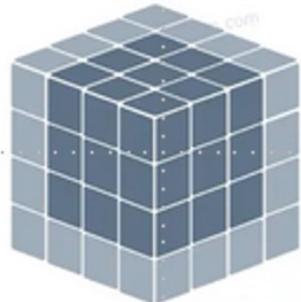


- Quatre catégories d'opérations selon l'objectif
- Agréger ou naviguer dans les hiérarchies  
**Roll-Up , Drill-Downd**
  - Changer les points de vue (axes)  
**Rotate/pivot, Switch , Split, nest, push, pull**
  - Sélectionner des sous-ensembles de données  
**Slice, dice, selection, projection**
  - Combiner plusieurs cubes  
**jointure (drill-across)**

## 5.1 – Agréger ou naviguer dans les hiérarchies

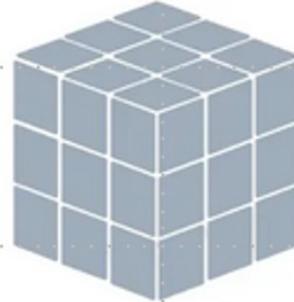
*Opérations de changement de niveau de détail*

Moins de détails



Roll-up

Plus de détails



Drill-down



# Roll-Up (Agréger)

- Le Roll-Up permet d'effectuer des agrégations (somme, moyenne, ...) en montant dans la hiérarchie d'une dimension
  - Agréger les données à un niveau supérieur
- Exemple :
  - Passer des Données par jour à des données par mois

Données  
par jour

Date (Jour)	Bateau	Skipper	Revenu Total (€)	Nombre de Réservations
2024-01-01	Alpha	Pierre	1,500	25
2024-01-02	Alpha	Pierre	1,800	30
2024-01-01	Beta	Jean	1,200	20



Roll-Up  
→

Date (Mois)	Bateau	Skipper	Revenu Total (€)	Nombre de Réservations	Données par mois
Janvier 2024	Alpha	Pierre	3,300	55	
Janvier 2024	Beta	Jean	1,200	20	

# Drill-Down (Détailler)

- Inverse du Roll-Up
- Permet de détailler des données en descendant dans la hiérarchie d'une dimension
  - Les mesures sont plus précises (plus granulaires)
- Exemple :
  - Passer des Données par mois à des données par jour



Date (Mois)	Bateau	Skipper	Revenu Total (€)	Nombre de Réservations
Janvier 2024	Alpha	Pierre	3,300	55
Janvier 2024	Beta	Jean	1,200	20

Données  
par mois

Drill-Down

Date (Jour)	Bateau	Skipper	Revenu Total (€)	Nombre de Réservations
2024-01-01	Alpha	Pierre	1,500	25
2024-01-02	Alpha	Pierre	1,800	30
2024-01-01	Beta	Jean	1,200	20

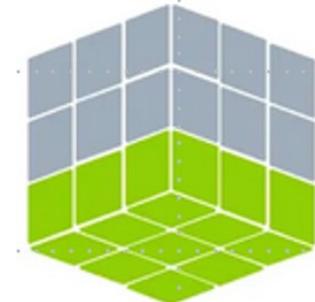
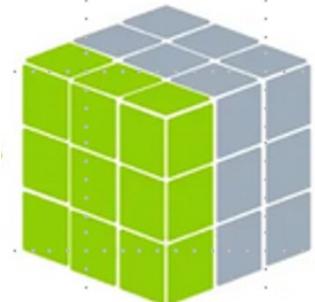
Données  
par jour

## 5.2 - Changer les points de vue

### *Opérations de changements de perspectives*

Réorganiser ou de restructurer le cube pour explorer les données de manière plus intuitive et pertinente

- **Rotate/Pivot** : Réorganiser les axes du cube
- **Switch** : Échanger deux dimensions
- **Split** : Séparer une dimension en sous-dimensions
- **Nest** : Imbriquer une dimension dans une autre
- **Push/Pull** : Ajouter/Supprimer une dimension

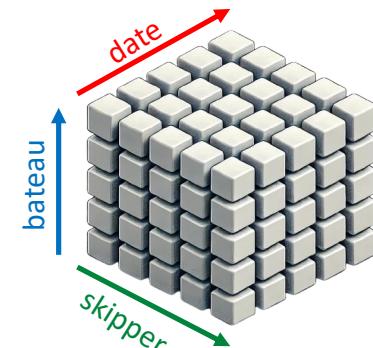


# Rotate/Pivot

- Réorganiser les axes pour observer les données sous un autre angle (faire "tourner le cube")
- **Exemple** Avant : Axe X = Skipper, Axe Y = Bateau, Axe Z= Date (ici 2024-01-01)  
Après : Axe X = Date , Axe Y = Skipper, Axe Z = Bateau

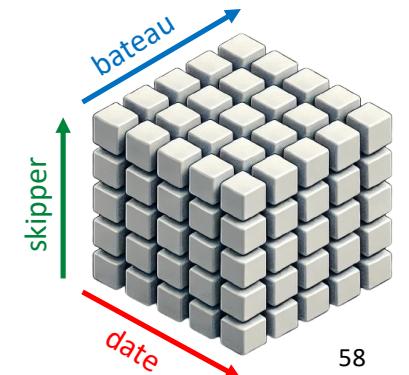
Revenus du 24-01-01 pour chaque skipper par bateau

Skipper	Bateau Alpha (2024-01-01)	Bateau Beta (2024-01-01)
Pierre	1,500 €	1,200 €
Jean	1,800 €	1,300 €



Revenus du 24-01-01 pour chaque bateau par skipper

Date	Bateau Alpha (Pierre)	Bateau Beta (Jean)
2024-01-01	1,500 €	1,200 €
2024-01-02	1,800 €	1,300 €



Rotate

# Switch (Echange)

- Échanger deux dimensions pour explorer la relation entre elles
  - Comparer les performances des skippers sur différentes périodes
- Exemple :
  - Avant : Axe X = **Date**, Axe Y = **Skipper**
  - Après : Axe X = **Skipper**, Axe Y = **Date**



Skipper	2024-01-01	2024-01-02
Pierre	1,500 €	1,800 €
Jean	1,200 €	1,300 €

Revenus quotidiens par skipper

Date	Skipper Pierre	Skipper Jean
2024-01-01	1,500 €	1,200 €
2024-01-02	1,800 €	1,300 €

Revenus par jour pour chaque skipper



# Split (Division)

- Découpe une dimension en nouvelles sous-dimensions
- Exemple :

Créer une hiérarchie dans la dimension date

- Avant : Une dimension unique Date
- Après : 3 sous dimensions Année, Mois, Jour,

Revenus quotidiens par date

Date	Revenu Total (€)
2024-01-01	2,700 €
2024-01-02	3,100 €



Année	Mois	Jour	Revenu Total (€)
2024	Janvier	01	2,700 €
2024	Janvier	02	3,100 €

# Nest (Imbrication)

- Imbriquer une dimension dans une autre pour créer une structure hiérarchique
- Exemple :
  - Imbriquer bateau dans skipper :
    - Chaque skipper sera associé à ses bateaux
    - Les bateaux ne seront explorés que par skipper



Skipper	Bateau	Revenu Total (€)	Dimensions indépendantes
Pierre	Alpha	1,500 €	
Pierre	Beta	1,200 €	
Jean	Alpha	1,800 €	
Jean	Beta	1,300 €	

Dimensions imbriquées

Nest

Skipper	Bateau Alpha	Bateau Beta
Pierre	1,500 €	1,200 €
Jean	1,800 €	1,300 €

# Push/Pull (Ajout/Retrait)

- Ajouter (push) ou retirer (pull) une dimension d'un axe pour modifier l'analyse
- Exemple :
  - Ajouter la dimension **Destination** pour analyser les revenus par destination
  - Retirer Bateau de l'axe Y pour une vue simplifiée



## 5.3 - Sélectionner des sous-ensembles de données

### *Opération de Filtrage-Extraction*

Pour extraire des sous-ensembles ou filtrer les données

- **Slice et Dice** : Extraire une tranche ou un sous-ensemble multidimensionnel
- **Selection** : Appliquer des critères pour filtrer des données spécifiques
- **Projection** : Réduire les dimensions ou les mesures pour se concentrer sur des données spécifiques

# Slice (Extraire une tranche)

- Fixe une valeur pour une dimension, réduisant ainsi le cube à un sous-cube d'une dimension en moins
- Exemple :
  - Analyser les revenus d'un jour spécifique

Avant : cube complet

Date	Bateau Alpha	Bateau Beta
2024-01-01	1,500 €	1,200 €
2024-01-02	1,800 €	1,300 €



Après (Fixation : Date= 2024-01-01)

Bateau	Revenu (€)
Alpha	1,500 €
Beta	1,200 €

# Dice (Extraire un sous-cube)

- Filtre les données selon plusieurs critères en combinant des valeurs pour une ou plusieurs dimensions ou mesures
- Exemple :
  - Critères : Date = Janvier 2024, Bateau = Alpha, Skipper=Pierre

Date	Bateau	Skipper	Revenu (€)	Nombre de Réservations
2024-01-01	Alpha	Pierre	1,500 €	20
2024-01-01	Alpha	Jean	1,200 €	15
2024-01-01	Beta	Pierre	1,800 €	25
2024-01-01	Beta	Jean	1,300 €	18
2024-01-02	Alpha	Pierre	1,700 €	22
2024-01-02	Alpha	Jean	1,400 €	19
2024-01-02	Beta	Pierre	1,900 €	28
2024-01-02	Beta	Jean	1,500 €	20

Sous-cube extrait selon les critères

Date	Bateau	Skipper	Revenu (€)	Nombre de Réservations
2024-01-01	Alpha	Pierre	1,500 €	20
2024-01-02	Alpha	Pierre	1,700 €	22

Dice



# Sélection

- Applique une condition de sélection (prédicat) pour inclure ou exclure certaines lignes
- Exemple :
  - Trouver les dates où les revenus des bateaux dépassent 1,500
  - Critères :  $\text{Revenu} > 1,500 \text{ €}$

Retourne un ensemble de données sans forcément créer de sous-cube structuré (comme Dice)

Selection  
→

Lignes sélectionnées selon les critères

Date	Bateau	Revenu (€)
2024-01-02	Alpha	1,800 €
2024-01-02	Beta	1,300 €

# Projection

- Réduit le cube en se concentrant sur certaines propriétés des dimensions ou certaines mesures
- Exemple :
  - Projeter année (dans Date) , nom (dans Bateau) et revenu

Date	Bateau	Skipper	Revenu (€)	Nombre de Réservations
2024-01-01	Alpha	Pierre	1,500 €	20
2024-01-01	Alpha	Jean	1,200 €	15
2024-01-02	Beta	Pierre	1,800 €	25
2024-01-02	Beta	Jean	1,300 €	18

Cube entier

Projection



Année	Bateau	Revenu (€)
2024	Alpha	2,700 €
2024	Beta	3,100 €

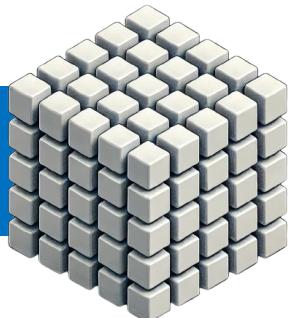
## 5.4 – Combiner plusieurs cubes

### Jointure (Drill-Across)

- Relie des cubes ayant des dimensions communes pour enrichir l'analyse
- Exemple :

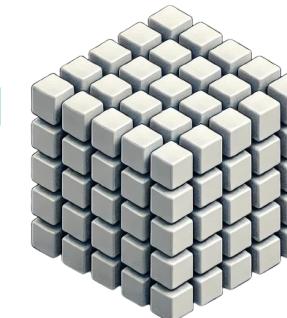
Cube 1 :

- Dimensions : Date, Bateau
- Mesure : Revenu



### Drill-Across

Jointure sur les dimensions Date et Bateau



Cube 2 :

- Dimensions : Date, Bateau
- Mesure : Coût

Date	Bateau	Revenu (€)	Coût (€)	Profit (€)
2024-01-01	Alpha	1,500 €	800 €	700 €
2024-01-01	Beta	1,200 €	600 €	600 €
2024-01-02	Alpha	1,800 €	900 €	900 €
2024-01-02	Beta	1,300 €	700 €	600 €

Nouvelle Mesure calculée  
 $\text{Profit} = \text{Revenu} - \text{Coût}$



## 6 – Interroger un cube à l'aide du Langage MDX

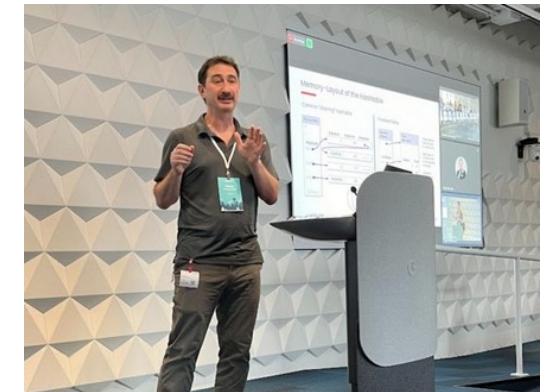
MultiDimensional EXpressions



# Qu'est-ce que le langage MDX ?

- MultiDimensional Expressions
- Langage d'interrogation conçu pour interagir avec des hypercubes OLAP
  - Exprimer des opérations sur des cubes (analyses complexes)
    - Drill-down, roll-up, slice, dice, ...
    - calculs d'aggregats (sommes, moyennes, ...)
- Comparable à SQL, mais adapté à la navigation dans des données multidimensionnelles
  - Plus puissant pour l'OLAP

Inventé par Mosha Pasumansky  
(Microsoft) en 1997



# MDX versus SQL

## SQL

- Données relationnelles :
  - tables, lignes, colonnes

```
SELECT colonnes  
FROM table  
WHERE condition
```

## MDX

- Données multi-dimensionnelles :
  - axes, dimensions, mesures

```
SELECT .... ON COLUMNS  
..... ON ROWS  
FROM cube  
WHERE <specification slice>
```

Attention, le where n'a pas la même signification en MDX

# MDX versus SQL

## SQL

- Table
  - Colonne
  - Plusieurs colonnes liées  
ou une table de dimension
  - Colonne
  - Valeur (d'une colonne)
- 
- Cube
  - Niveau (level)
  - Dimension
  - Mesure
  - Membre de dimension

## Explorateur du cube

The screenshot shows the IcCube MDX interface. On the left, there is a sidebar titled "Explorateur du cube" which lists various cube objects like DWcruises, CubeDateBateauxSkippers, bateau, date, destination, ville, id\_destination, All, Corse, Bastia, Ajaccio, Propriano, PACA, and Cannes. A blue arrow points from the top-left towards the MDX editor area. The main area has tabs for "MDX" and "search for ...". The "MDX" tab is active, showing the following content:

**Editeur MDX**

```
SELECT
    [Measures].[Revenu Total] ON COLUMNS,
    [destination].[ville].Members ON ROWS
FROM [CubeDateBateauxSkippers]
```

**Résultat de la requête**

	Revenu Total
Bastia	8,526.53
Ajaccio	16,017.30
Propriano	39,797.14
Cannes	21,926.56

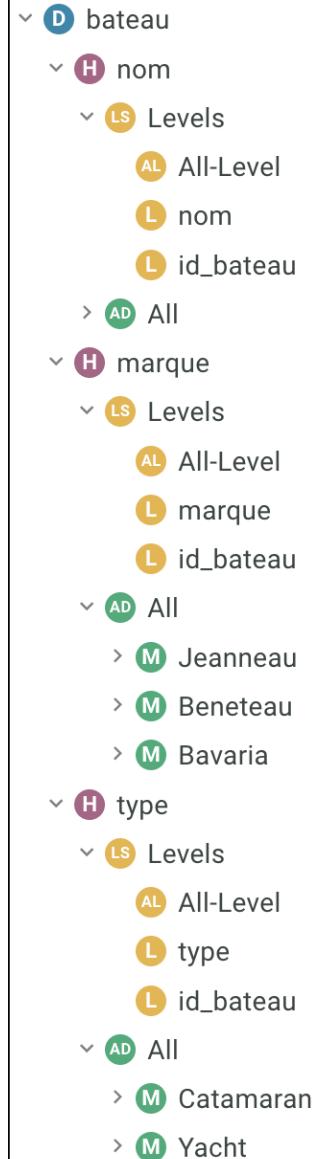
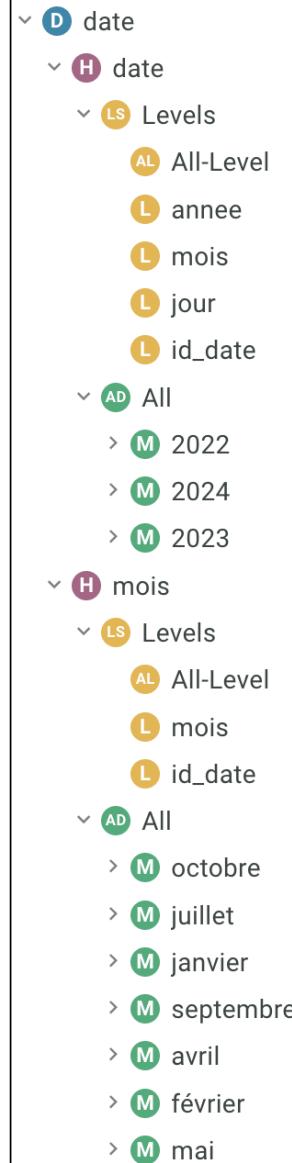
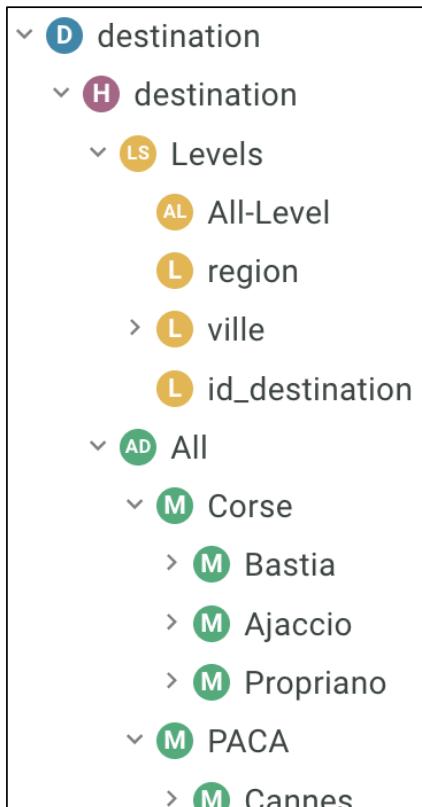
# Cube exemple

## C CubeCruises

### MD Measures

NbReservations

RevenuTotal



# Structure générale d'une requête MDX

SELECT

Définit les axes du cube résultat

[<specification axe1>  
[, <spécification axe2>... ]]

En théorie, n  
axes, mais en  
général 2 pour  
être  
visualisables

FROM

<Spécification Cube (s) > Définit le(s) cube(s) Source(s)

[WHERE

[<spécification d'un filtre (slicer)> ]

Filtres sur le(s) cube(s) source(s)

# Description d'un axe associé à une dimension

nom de la dimension

ex: [bateau]

une valeur ou un élément particulier dans la hiérarchie  
ex. : [Yacht], [Catamaran]

[**Dimension**] . [**Hiérarchie**] . [**Membre**] ON COLUMNS

Niveau hiérarchique ou attribut dans la dimension  
ex. : [bateau].[marque]

Exemples de spécification d'axes liés à une dimension

```
-- Un type de bateau spécifique  
[bateau].[type].[Catamaran]  
-- tous les membres d'un niveau  
[bateau].[type].Members  
-- le premier membre d'un niveau  
[destination].[Region].FirstChild
```

- ON ROWS  
les axes peuvent aussi être numérotés
  - ON 0
  - ON 1

# Description d'un axe associé à une mesure

mot clé

nom de la mesure  
ex. : [Revenu Total]

[**Measures**] . [*nom mesure*] ON COLUMNS

Exemples de spécification d'axes liés à une mesure

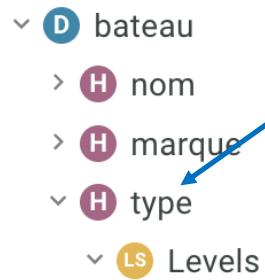
```
--la mesure RevenuTotal  
[Measures] . [RevenuTotal]  
--la mesure NbReservations  
[Measures] . [NbReservations]
```

- ON ROWS  
les axes peuvent aussi être numérotés
  - ON 0
  - ON 1

# En général 2 axes : COLUMNS (axe 0) et ROWS(axe 1)

- Exemple : Afficher le revenu total par type de bateaux

```
SELECT  
    [Measures] . [RevenuTotal] ON COLUMNS,  
    [type] . [type] .Members ON ROWS  
FROM [CubeCruises]
```



choix de la hierarchie  
mois

	RevenuTotal
> Catamaran	4,867.02
> Yacht	81,400.51

# Equivalent en SQL (sur le DW)

- Exemple : Afficher le revenu total par type de bateaux

```
SELECT  
    [Measures].[RevenuTotal] ON COLUMNS,  
    [type].[type].Members ON ROWS  
FROM [CubeCruises]
```

MDX

En MDX les  
SUM sont  
précalculées

```
SELECT b.type AS type_bateau,  
      SUM(f.revenu_total) AS revenu_total  
FROM  
      Faits_Ventes f  
JOIN  
      Dimension_Bateau b ON f.id_bateau = b.id_bateau  
GROUP BY b.type;
```

SQL

# Exemple avec un membre spécifique

- Afficher le revenu total pour le mois d'octobre (toutes années confondues)

```
SELECT  
    [Measures].[RevenuTotal]    ON COLUMNS,  
    [date].[mois].[octobre]     ON ROWS  
FROM [CubeCruises]
```



	RevenuTotal
➤ octobre	37,594.07

# Autres méthodes ou fonctions spécifiques

## ■ .Members

- désigne tous les membres (toutes les valeurs) d'une dimension ou d'un niveau

[Dimension] . [Hiérarchie] . **Members**

## ■ .Children

- désigne tous les enfants d'un membre spécifique dans une hiérarchie
- Ex: tous les mois de l'année 2023 (car mois est l'enfant de année)

[date] . [annee] . [2023] . **Children**

## ■ .Parent

- désigne le parent d'un membre spécifique dans une hierarchie
- Ex: toutes les années qui ont des croisieres au mois d'octobre (car annee est le parent de mois)

[date] . [mois] . [octobre] . **Parent**

# Requêtes avec des set - ensembles {}

- {<specification >} : un set explicite
  - permet de lister un ensemble de membres, de tuples ou d'éléments
- Exemple : Affiche les revenus pour les bateaux de type Yacht

```
SELECT  
    [Measures] . [RevenuTotal] ON COLUMNS,  
    { [bateau] . [type] . [Catamaran] , [bateau] . [type] . [Yacht] }  
    ON ROWS  
FROM [CubeCruises]
```



	Revenu Total
> Catamaran	4,867.02
> Yacht	81,400.51

# Exemple avec filtrage (where)

- Calculer le Revenu total pour l'année 2023

```
SELECT  
    [Measures] . [RevenuTotal]  ON COLUMNS  
FROM  [CubeCruises]  
WHERE  ([date] . [annee] . [2023])
```

SLICER : le Where restreint le cube en fixant ici la dimension date



	2023
Revenu Total	22,139.09

Attention le where n'exprime pas une condition comme en SQL

# Exemple avec Drill-down

- Analyser les revenus totaux par année et par mois

```
SELECT  
    [Measures].[RevenuTotal] ON COLUMNS,  
    DrillDownLevel([date].[année].Members)  
    ON ROWS  
FROM [CubeCruises]
```



Renvoie les détails de chaque membre en dessous de [année] : descente d'un niveau (mois ici)

	Revenu Total
▼ 2022	47,033.12
➤ octobre	32,144.04
➤ septembre	6,362.55
➤ avril	8,526.53
▼ 2024	17,095.32
➤ octobre	5,450.03
➤ janvier	2,786.08
➤ mai	8,859.21
▼ 2023	22,139.09
➤ juillet	4,867.02
➤ janvier	7,617.32
➤ février	9,654.75

# Exemple avec un calcul de maximum TopCount

- Trouver le mois avec le revenu maximum

SELECT

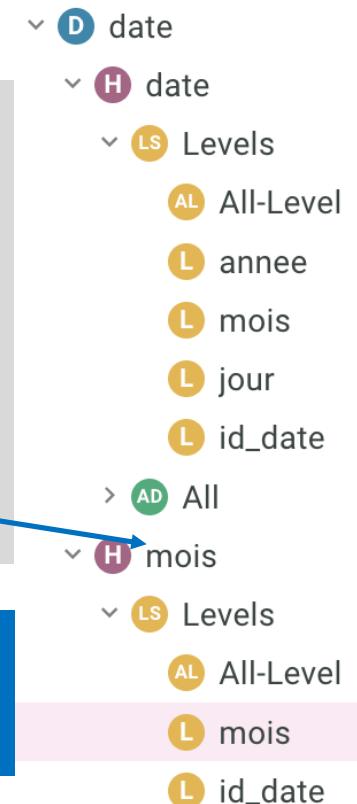
```
[Measures] . [RevenuTotal] ON COLUMNS,  
TopCount (  
    [mois] . [mois] . Members, 1,  
    [Measures] . [Revenu Total]  
) ON ROWS  
FROM [CubeCruises]
```

Si n: on obtient les n premiers mois



	RevenuTotal
> octobre	37,594.07

choix de la hierarchie  
mois



# Exemple avec calculs **With Member**

- Calculer le revenu moyen par réservation

```
WITH MEMBER [Measures]. [Revenu Moyen par Réservation] AS  
    [Measures]. [RevenuTotal] /  
    [Measures]. [NbReservations]  
  
SELECT  
    [Measures]. [Revenu Moyen par Réservation] ON COLUMNS,  
    [type]. [type].Members ON ROWS  
  
FROM [CubeCruises]
```

C'est comme si l'on ajoutait  
une mesure calculée à  
partir des autres



	Revenu Moyen ...
> Catamaran	4,867.02
> Yacht	2,806.91

# Exemple avec imbrications (nest)

- Afficher les revenus totaux pour chaque nom de skipper en détaillant les noms de bateaux

```
SELECT  
    [skipper].[nom].Members *  
    [nom].[nom].Members ON ROWS,  
    [Measures].[RevenuTotal]  
        ON COLUMNS  
FROM [CubeCruises]
```



Comment éliminer  
les cases vides ?

		RevenuTotal
➤ Jean Dupont	➤ Le Grand Bleu	
	➤ Horizon	
	➤ Étoile du Ma...	
	➤ Vent du Large	
	➤ Océan Sauv...	
➤ Luc Martin	➤ Le Grand Bleu	
	➤ Horizon	
	➤ Étoile du Ma...	
	➤ Vent du Large	
	➤ Océan Sauv...	
➤ Pierre Durand	➤ Le Grand Bleu	
	➤ Horizon	2,786.08
	➤ Étoile du Ma...	9,654.75
	➤ Vent du Large	
	➤ Océan Sauv...	
➤ Antoine Lecl...	➤ Le Grand Bleu	
	➤ Horizon	7,617.32
	➤ Étoile du Ma...	8,526.53
	➤ Vent du Large	
	➤ Océan Sauv...	

# Exemple avec imbrications (nest)

- Afficher les revenus totaux pour chaque nom de skipper en détaillant les noms de bateaux

```
SELECT  
    NON EMPTY  
        [skipper].[nom].Members *  
        [nom].[nom].Members ON ROWS,  
        [Measures].[RevenuTotal]  
    ON COLUMNS  
FROM [CubeCruises]
```



Elimination des  
lignes vides

		RevenuTotal
> Pierre Durand	> Horizon	2,786.08
	> Étoile du Ma...	9,654.75
> Antoine Lecl...	> Horizon	7,617.32
	> Étoile du Ma...	8,526.53
> Thomas Gar...	> Vent du Large	4,797.61
> Victor Lefevre	> Horizon	6,362.55
> Olivier Bern...	> Vent du Large	5,450.03
> Marc Petit	> Étoile du Ma...	36,205.64
> Philippe Rob...	> Le Grand Bleu	4,867.02

# Autre solution équivalente : **CrossJoin**

- Afficher les revenus totaux pour chaque nom de skipper en détaillant les noms de bateaux

```
SELECT  
    NON EMPTY  
    CROSSJOIN(  
        [skipper].[nom].Members ,  
        [bateau].[nom].[nom].Members)  
    ON ROWS,  
    [Measures].[RevenuTotal]  
    ON COLUMNS  
FROM [CubeCruises]
```

jointure entre les dimensions skipper et bateau



		RevenuTotal
> Pierre Durand	> Horizon	2,786.08
	> Étoile du Ma...	9,654.75
> Antoine Lecl...	> Horizon	7,617.32
	> Étoile du Ma...	8,526.53
> Thomas Gar...	> Vent du Large	4,797.61
> Victor Lefevre	> Horizon	6,362.55
> Olivier Bern...	> Vent du Large	5,450.03
> Marc Petit	> Étoile du Ma...	36,205.64
> Philippe Rob...	> Le Grand Bleu	4,867.02

# Exemple avec Condition IIF

- Afficher le nombre de réservations par nom de bateau en affichant 0 si le bateau n'a aucune réservation

```
WITH
    MEMBER [Measures]. [Nombre Reservations]
AS
    IIF(IsEmpty([Measures]. [NbReservations]),
        0, [Measures]. [NbReservations])
SELECT
    [bateau].[nom].[nom].Members ON ROWS,
    [Measures]. [Nombre Reservations] ON
COLUMNS
FROM [CubeCruises]
```

	Nombre Reserv...
> Étoile du Matin	16.00
> Horizon	7.00
> Vent du Large	6.00
> Le Grand Bleu	1.00
> Océan Sauvage	0

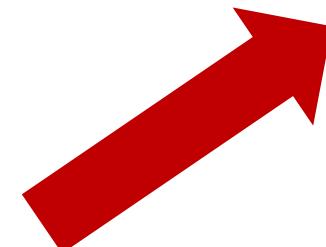


# Exemple avec Filtrage sur une dimension

- Afficher les revenus totaux pour chaque année en se limitant aux années ayant totalisé un revenu inférieur à 25000 euros

```
SELECT  
    FILTER(  
        [date] . [annee] .Members,  
        [Measures] . [RevenuTotal] < 25000  
    ) ON ROWS,  
    [Measures] . [RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

	RevenuTotal
> 2024	17,095.32
> 2023	22,139.09



# Exemple avec Filtrage sur une dimension et crossjoin

- Afficher les revenus totaux pour chaque année en se limitant aux années ayant totalisé un revenu inférieur à 25000 en détaillant les noms des bateaux associés

```
SELECT  
    CROSSJOIN( FILTER(  
        [date].[annee].Members,  
        [Measures].[RevenuTotal] < 25000  
    ) , [bateau].[nom].[nom].members)  
ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

	RevenuTotal
> 2024	> Le Grand Bleu
	> Horizon 2,786.08
	> Étoile du Ma... 8,859.21
	> Vent du Large 5,450.03
	> Océan Sauv...
> 2023	> Le Grand Bleu 4,867.02
	> Horizon 7,617.32
	> Étoile du Ma... 9,654.75
	> Vent du Large
	> Océan Sauv...

Comment Modifier la requête pour que les bateaux sans revenus ne soient pas affichés

# Solution : ajouter NON EMPTY

```
SELECT  
    NON EMPTY CROSSJOIN( FILTER(  
        [date].[annee].Members,  
        [Measures].[RevenuTotal] < 25000  
    ) , [bateau].[nom].[nom].members) ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```



		RevenuTotal
➤ 2024	➤ Horizon	2,786.08
	➤ Étoile du Matin	8,859.21
	➤ Vent du Large	5,450.03
➤ 2023	➤ Le Grand Bleu	4,867.02
	➤ Horizon	7,617.32
	➤ Étoile du Matin	9,654.75

# Exemple de requête avec somme **Sum**

- Afficher les revenus totaux par skipper, mais uniquement pour les années où le skipper a réalisé plus de 10 000 €

```
SELECT  
    FILTER(  
        [skipper] . [nom] .Members,  
        SUM([date] . [annee] .Members,  
            [Measures] . [RevenuTotal]) > 10000  
    ) ON ROWS,  
    [Measures] . [RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

	RevenuTotal
➤ Marc Petit	36,205.64
➤ Antoine Leclerc	16,143.85
➤ Pierre Durand	12,440.83



# Exemple de requête avec somme (modifiée)

- Afficher les revenus totaux par skipper **et par année**, mais uniquement pour les années où le skipper a réalisé plus de 10 000 €

```
SELECT  
NON EMPTY CROSSJOIN(FILTER(  
[skipper].[nom].Members,  
SUM([date].[annee].Members, [Measures].[RevenuTotal]) > 10000)  
, [date].[annee].Members ) ON ROWS,  
[Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```



		RevenuTotal
➤ Pierre Durand	➤ 2024	2,786.08
	➤ 2023	9,654.75
➤ Antoine Lecl...	➤ 2022	8,526.53
	➤ 2023	7,617.32
➤ Marc Petit	➤ 2022	27,346.43
	➤ 2024	8,859.21

# Exemple de requête avec comparaisons

- Afficher pour chaque skipper, ses revenus totaux pour 2023 et 2024 ainsi que la différence entre les deux



	Revenu2023	Revenu2024	Variation
➤ Jean Dupont	0	0	0
➤ Luc Martin	0	0	0
➤ Pierre Durand	9,654.75	2,786.08	-6,868.67
➤ Antoine Leclerc	7,617.32	0	-7,617.32
➤ Thomas Garnier	0	0	0
➤ Charles Blanc	0	0	0
➤ Victor Lefevre	0	0	0
➤ Olivier Bernard	0	5,450.03	5,450.03
➤ Marc Petit	0	8,859.21	8,859.21
➤ Philippe Robert	4,867.02	0	-4,867.02

# Exemple de requête avec comparaisons

- Afficher pour chaque skipper, ses revenus totaux pour 2023 et 2024 ainsi que la différence entre les deux

```
WITH MEMBER [Measures].[Revenu] AS
    IIF(IsEmpty([Measures].[RevenuTotal]), 0, [Measures].[RevenuTotal])
    MEMBER [Measures].[Revenu2023] AS
        ([Measures].[Revenu], [date].[annee].[2023])
    MEMBER [Measures].[Revenu2024] AS
        ([Measures].[Revenu], [date].[annee].[2024])
    MEMBER [Measures].[Variation] AS
        [Measures].[Revenu2024] - [Measures].[Revenu2023]
```

On ajoute des mesures calculées à partir des mesures

```
SELECT
    [skipper].[nom].Members ON ROWS,
    { [Measures].[Revenu2023], [Measures].[Revenu2024],
        [Measures].[Variation] }
    ON COLUMNS
FROM [CubeCruises]
```

# A vous de jouer (4) – Requêter Cubecruises

Définissez en MDX les requêtes suivantes sur le cube **Cubecruises** :

1. Afficher le revenu total par nom de bateau
2. Afficher le nombre de réservations par nom de skipper
  - Version2 : afficher 0 si le skipper n'a aucune réservation
3. Afficher le revenu total et le nombre de réservations par marque de bateau
4. Afficher les skippers avec un revenu total supérieur à 10 000 €
5. Afficher les revenus par skipper pour l'année 2023
  - Version 2 : restreindre l'affichage aux skippers ayant eu effectivement des revenus en 2023
6. Afficher les revenus totaux par année et par type de bateau
7. Afficher les revenus totaux par skipper et par nom de bateau
  - Version 2 : par nom de bateau et par skipper
8. Afficher les noms des deux bateaux qui ont généré le plus de revenu et préciser leurs revenus



# A vous de jouer (5)..

- Réécrivez ces requêtes en SQL en interrogeant directement le datawarehouse DWCruisesEtoile.



- Vous pouvez vous aider d'un assistant ...



- Qu'en pensez-vous ?

Intérêts de MDX?



# Correction

--1 Afficher le revenu total par nom de bateau

```
SELECT
    [bateau].[nom].Members ON ROWS,
    [Measures].[RevenuTotal] ON COLUMNS
FROM [CubeCruises]
```

--2 Afficher le nombre de réservations par nom de skipper

```
SELECT
    [skipper].[nom].Members ON ROWS,
    [Measures].[NbReservations] ON COLUMNS
FROM [CubeCruises]
```

--solution pour afficher 0 si vide

```
WITH
    MEMBER [Measures].[NbReservationsFixed] AS
        IIF(IsEmpty([Measures].[NbReservations]), 0,
            [Measures].[NbReservations])
```

```
SELECT
    [skipper].[nom].AllMembers ON ROWS,
    [Measures].[NbReservationsFixed] ON COLUMNS
FROM [CubeCruises]
```

--3 Afficher le revenu total et le nombre de réservations par marque de bateau

```
SELECT
    [bateau].[marque].Members ON ROWS,
    {[Measures].[RevenuTotal], [Measures].[NbReservations]} ON COLUMNS
FROM [CubeCruises]
```

--4 Afficher les skippers générant un revenu total supérieur à 10 000 €

```
WITH
    MEMBER [Measures].[RevenuTotalFixed] AS
        IIF(IsEmpty([Measures].[RevenuTotal]), 0, [Measures].[RevenuTotal])
SELECT
    FILTER(
        [skipper].[nom].Members,
        [Measures].[RevenuTotalFixed] > 10000
    ) ON ROWS,
    [Measures].[RevenuTotalFixed] ON COLUMNS
FROM [CubeCruises]
```

# Correction

--5 Afficher les revenus des skippers pour l'année 2023

```
SELECT  
    [Measures].[RevenuTotal] ON COLUMNS,  
    [skipper].[nom].Members ON ROWS  
FROM [CubeCruises]  
WHERE  
    [date].[annee].[2023]
```

-- version qui affiche uniquement les skippers ayant eu un revenu en 2023

```
SELECT  
    NON EMPTY  
        [skipper].[nom].Members ON ROWS,  
        [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

--6 Afficher les revenus totaux par année et par type de bateau

```
SELECT  
    CrossJoin([date].[annee].Members, [bateau].[type].Members) ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

-- autre solution

```
SELECT  
    [date].[annee].Members * [bateau].[type].Members ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```

-- 7 Afficher les revenus totaux par skipper et par nom de bateau

```
SELECT  
    CrossJoin([skipper].[nom].Members, [bateau].[nom].Members) ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]  
-- en sens inverse  
SELECT  
    CrossJoin([bateau].[nom].Members, [skipper].[nom].Members) ON ROWS,  
    [Measures].[RevenuTotal] ON COLUMNS  
FROM [CubeCruises]
```



# Correction

--8 afficher les noms des deux bateaux qui ont généré le plus de revenu et préciser leurs revenus

```
SELECT  
    [Measures].[RevenuTotal] ON COLUMNS,  
    TopCount(  
        [bateau].[nom].[nom].Members, 2,[Measures].[RevenuTotal]  
    ) ON ROWS  
FROM [CubeCruises]
```



# A vous de jouer (6) - – Requêter DWCruisesPerf

- Définissez en MDX les requêtes suivantes sur votre cube :

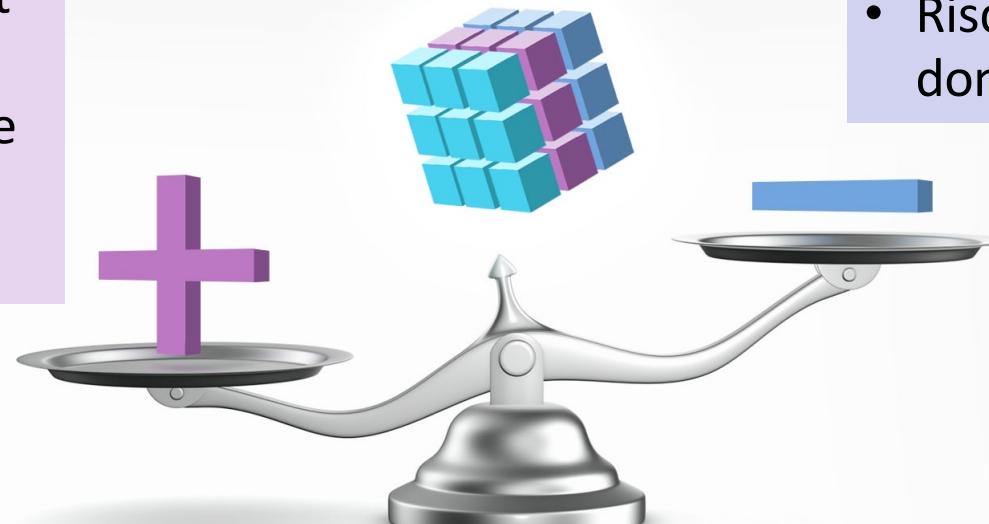
1. Quelle est la durée moyenne pour l'ensemble des croisières ?
2. Combien de passagers ont été transportés par chaque bateau ?
3. Quels sont les deux bateaux qui ont transporté le plus de passagers?
4. Quelle a été la durée moyenne des croisières pour chaque bateau ?
5. Quelle est la durée moyenne des croisières gérées par chaque skipper ?
6. Quel est le nombre total de passagers embarqués par chaque skipper ?
7. Quel est le nombre total de passagers embarqués par niveau d'expérience des skippers ?
  - $0 < \text{expérience} < 10$  : débutant -  $10 \leq \text{expérience} < 20$  : intermédiaire -  $20 \leq \text{expérience}$  : expert
8. Quelle est le nombre de passager par skipper pour chaque bateau?



# Pour ou contre OLAP ?

- Représentation multidimensionnelle  
-> analyse sous plusieurs angles
- Rapidité de traitement et de conception
- Navigation fluide entre données détaillées et agrégées

- Coût élevé mais rentable à long terme
- Limites computationnelles
- Risques liés aux données limitées



source image : <https://galaktika-soft.com/blog/olap-cubes.html>

# OLAP, Dataviz ou Excel ?

Scénario	OLAP	Outils de DataViz	Excel
<b>Explorer des hiérarchies complexes</b>	Idéal	Peut afficher mais pas naviguer dynamiquement	Possible mais limité via PivotTables
<b>Présenter des données de manière visuelle</b>	Limité (pas son rôle principal)	Idéal : graphiques interactifs	Suffisant pour des présentations simples
<b>Analyser des données volumineuses</b>	Très performant	Dépend de la source connectée	Lent et peu évolutif
<b>Travail individuel ou rapide</b>	Nécessite une infrastructure et un outil dédié	Peut convenir si les données sont prêtes	Idéal pour les analyses rapides et simples

# Liens

- Les excellents cours de Bernard Espinasse
  - OLAP <https://pageperso.lis-lab.fr/bernard.espinasse/wp-content/uploads/2021/12//3-OLAP-4p.pdf>
- Un article sur les concepts OLAP qui met les concepts en perspective
  - <https://grim.developpez.com/articles/concepts/olap/>