

# TP : Introduction aux Entrepôts de données

- **Objectifs.** L'objectif de ce TP est de consolider les notions de base utilisées dans le cadre des entrepôts de données : une étape fondamentale pour développer d'autres thèmes dans la suite du module.
- **Travail demandé.** Le travail personnel consiste à adresser l'ensemble des questions de ce document, ainsi qu'à rédiger vos réponses de façon claire et concise dans un document (word, latex, ou équivalent) que vous devrez rendre dans Moodle. Ne sous-estimez pas l'utilité de la rédaction, qui vous aide à conceptualiser et structurer vos connaissances.
- **Utilisation IA.** L'utilisation des IA pour la résolution des questions est déconseillée, car l'intérêt du TP est de prendre du temps pur réfléchir aux questions, discuter avec son binôme et l'enseignant, afin d'arriver à assimiler des concepts qui sont *nouveaux* (e.g., discerner requêtes analytiques et transactionnelles, définir un modèle en étoile). Cet intérêt se dilue si l'IA le fait pour nous ... et même l'IA n'est pas à l'abri des erreurs ;-). L'approche conseillée en général est d'utiliser l'IA pour accélérer notre travail lorsqu'on est en mesure de pouvoir **vérifier** les réponses qu'elle nous fournit, ce qui requiert d'abord de familiariser avec le sujet.

## 1) Les interrogations : requêtes transactionnelles vs analytiques

Considérons le cas d'un système d'information d'une chaîne nationale de cinémas multisalles.

Pour chacune des requêtes suivantes, identifiez s'il s'agit d'une requête transactionnelle ou analytique. Justifiez vos réponses, en faisant des hypothèses si nécessaire.

1. Restent-ils des billets à Montpellier pour la séance de 20 heures du film *Logan* ?
2. Aurait-on éventuellement pu proposer des plus grandes salles et plus de séances pour le dernier film de *Star Wars* ?
3. (Traduire la requête suivante en langage naturel, puis indiquer s'il s'agit d'une requête transactionnelle ou analytique)

```
SELECT Film.titre, Cinema.nom, Date.mois, COUNT(Place.placeID)
FROM Film, Ventes, Cinema, Place, Temps, Date
WHERE Ventes.filmID = Film.filmID
      AND Ventes.cinemaID = Cinema.cinemaID
      AND Ventes.tempsID = Temps.tempsID
      AND Place.cinemaID = Cinema.cinemaID
      AND Ventes.dateID = Date.dateID
GROUP BY Film.titre, Cinema.nom, Date.mois;
```

4. (Traduire la requête suivante en langage naturel, puis indiquer s'il s'agit d'une requête transactionnelle ou analytique)

```
SELECT Temps.creneau, COUNT(*)
FROM Ventes, Temps
WHERE Ventes.tempsID = Temps.tempsID
GROUP BY Temps.creneau;
```

5. (Traduire la requête suivante en langage naturel, puis indiquer s'il s'agit d'une requête transactionnelle ou analytique)

```
INSERT INTO Ventes
VALUES ('film1','cinema24','date2','temps3','place44','7.50');
```

---

## 2) Un entrepôt de données pour Amazon (<https://www.amazon.fr/>)

1. Proposez un modèle en étoile pour un entrepôt de données permettant l'analyse des ventes dans Amazon.  
Le modèle doit permettre une analyse temporelle des ventes en fonction des **produits**, des **utilisateurs**, ainsi que des **promotions** mises en place.
2. Comment feriez-vous pour intégrer les commentaires clients dans ce modèle ?
3. Proposez trois requêtes analytiques pour le modèle de données en étoile que vous avez conçu.

---

## 3) Requêtes Analytiques Monoprix

Considérez la table de faits (simplifiée) suivante, enregistrant les ventes journalières chez Monoprix :

```
CREATE TABLE ventes_monoprix (  
  id_date    VARCHAR(10) NOT NULL,  
  id_produit VARCHAR(10) NOT NULL,  
  id_magasin VARCHAR(10) NOT NULL,  
  id_ville   VARCHAR(10) NOT NULL,  
  montant_journalier NUMBER(10,2) NOT NULL  
);
```

Vous pouvez remplir la base à partir des données disponibles dans le fichier `DW_monoprix.sql` disponible dans le repertoire.

Exprimez en SQL les interrogations suivantes à l'aide de `GROUP BY` :

1. Donner le montant total des ventes par produit.
2. Donner le montant total des ventes par produit et par ville.
3. Donner le montant total des ventes par produit et par jour.
4. Donner la moyenne du montant des ventes par magasin et par jour.
5. Donner le montant total des ventes par ville et par jour.
6. Donner le montant total des ventes par produit, ville et jour.

Enfin, testez les options `ROLLUP` et `CUBE` et comparez les résultats. Pourrait-on regrouper les interrogations grâce à ces options ?

---

## 4) Classification des faits

Considérez la liste de faits ci-dessous.

- a. Précisez s'il s'agit d'un fait **transactionnel** ou d'un **snapshot**.
  - b. Précisez si la mesure est **additive**, **semi-additive** ou **non-additive**.
1. Un fait  $(j,p,c,m,x)$  existe lorsqu'un produit  $p$  est acheté par un client  $c$  le jour  $j$  au magasin  $m$ . La mesure  $x$  correspond au prix total.
  2. Un fait  $(j,p,m,x)$  existe lorsqu'un produit  $p$  est acheté le jour  $j$  au magasin  $m$ . La mesure correspond au chiffre d'affaires.
  3. Un fait  $(j,p,m,x)$  existe pour chaque combinaison de produit  $p$ , magasin  $m$  et jour  $j$ . La mesure  $x$  correspond au stock de  $p$  en  $m$  le jour  $j$ .
  4. Un fait  $(j,p,m,x)$  existe pour chaque combinaison de produit  $p$ , magasin  $m$  et jour  $j$ . La mesure  $x$  correspond au nombre de ventes de  $p$  en  $m$  cumulées depuis le début de l'année jusqu'au jour  $j$ .
  5. Un fait  $(c,e,j)$  existe lorsqu'un appel du client  $c$  le jour  $j$  est traité par l'employé  $e$ . Aucune mesure n'existe.
  6. Un fait  $(c,j,x)$  existe lorsqu'un client  $c$  le jour  $j$  laisse une note sur un produit acheté. La mesure  $x$  est la note donnée.
  7. Un fait  $(c,e,j,x)$  existe lorsqu'un appel du client  $c$  le jour  $j$  est traité par l'employé  $e$ . La mesure  $x$  est la durée de l'appel en secondes.

8. Un fait  $(m,b,j,x)$  existe lorsque la monnaie  $m$  est changée à la banque  $b$  le jour  $j$ . La mesure  $x$  est le montant total échangé en euros.
9. Un fait  $(m,b,j,x)$  existe lorsque la monnaie  $m$  est changée à la banque  $b$  le jour  $j$ . La mesure  $x$  est le cours de change moyen de  $m$  en euros pour toutes les transactions du jour  $j$ .