

GROUPE : JONATHAN WILD - YANIS STENTZEL

LUNDI 8  
DÉCEMBRE



# SAE 3.02 - INTÉGRATION DE DONNÉES DANS UN DATAWAREHOUSE

Entrepôt de Données GTS  
- Données financières

# NOTRE ÉQUIPE



YANIS STENTZEL JONATHAN WILD

# OBJECTIFS ET PROBLèMATIQUE

- Objectif SAE : Mise en œuvre d'un entrepôt de données (ROLAP) pour l'aide à la décision.

Sujet choisi : Gestion Financière.

Sources : Fichiers hétérogènes (CSV, XLSX, XML) sans base de données opérationnelle.

## Problématiques :

- Quel est le profil des personnes présentes dans cette base ? (Âge, localisation, etc.)
- Quelles sont les ventes mensuelles ? (Évolution du chiffre d'affaires)
- Qui sont les profils qui ont un paiement 2 mois consécutifs en erreur ?
- Quel est le titre le plus populaire ?

# ÉTAPE 1 - COMPRÉHENSION DES DONNÉES

2025

Création du dictionnaire des données et  
identification des liens inter-fichiers.

1. **FIN\_ORDER.XLSX**
2. **FIN\_GT.TXT**
3. **CUSTOMER.XML**



# DICTIONNAIRE DE DONNÉES

Entité	Variable	Description
Parent	CustomerID	Identifiant unique du Client/Parent. (Int)
	P_FirstName	Prénom du Parent.
	P_Birthdate	Année de naissance du Parent.
	P_Civility	Sexe du Parent. (H/F)
	CityHome	Ville de résidence.
Lien	Child_ID	Identifiant de l'Enfant.
	LastName	Nom de Famille (commun).
Enfant	FirstName	Prénom de l'Enfant.
	BirthDate	Date de naissance complète de l'Enfant. (JJ/MM/AAAA)
	AccountStatus	Statut du compte de l'enfant.
	Civility	Sexe de l'Enfant. (H/F)

Type de Donnée	Variable / Catégorie	Description
Produit (Titre)	Abo 1 an	Abonnement annuel.
	Abo 30 Jou	Abonnement mensuel.
	1 voyage	Billet unitaire.
	10 voyages	Abonnement pour 10 voyages.

Fin\_Order.xlsx : Ce classeur contient l'ensemble des données transactionnelles, décomposées en trois niveaux de granularité.

Feuille Order =

Variable	Description
Order	Identifiant de la commande. (nombre entier)
CustomerID	Identifiant du client qui a passé la commande. ("OD-" + Numéro (4 chiffres))
OrderDate	Date de passation de la commande. (JJ/MM/AAAA)
TotalAmount	Montant total de la commande.
OrderStatus	Statut actuel de l'en tête de commande. (INITIALIZED,PAID,UNPAID)
PaymentMode	Mode de paiement utilisé (CB, CA).

Feuille Order Schedules =

Variable	Description
OrderScheduleID	Identifiant unique du paiement. (ODS- + Numéro (4 chiffres))
OrderID	Identifiant de la commande. (OD- + Numéro (4 chiffres))
TransactionDate	Date de l'opération de paiement. (JJ/MM/AAAA)
TransactionStatus	Statut de l'opération de paiement. (OK, ERROR, TO_PAID)
RetryTime	Nombre de tentatives pour ce paiement.

Feuille Item =

Variable	Description
OrderItemID	Identifiant unique de la ligne d'article. (OI- + Numéro (4 chiffres))
Order	Identifiant de la commande. (OD- + Numéro (4 chiffres))
TotalAmount	Montant total de l'article commandé.
Quantity	Quantité de l'article.
UnitaryAmount	Montant unitaire de l'article.
Support	Identifiant du support (physique ou dématérialisé). (E + Année + Numéro (4 chiffres))

# ÉTAPE 2 - MODÉLISATION EN ÉTOILE

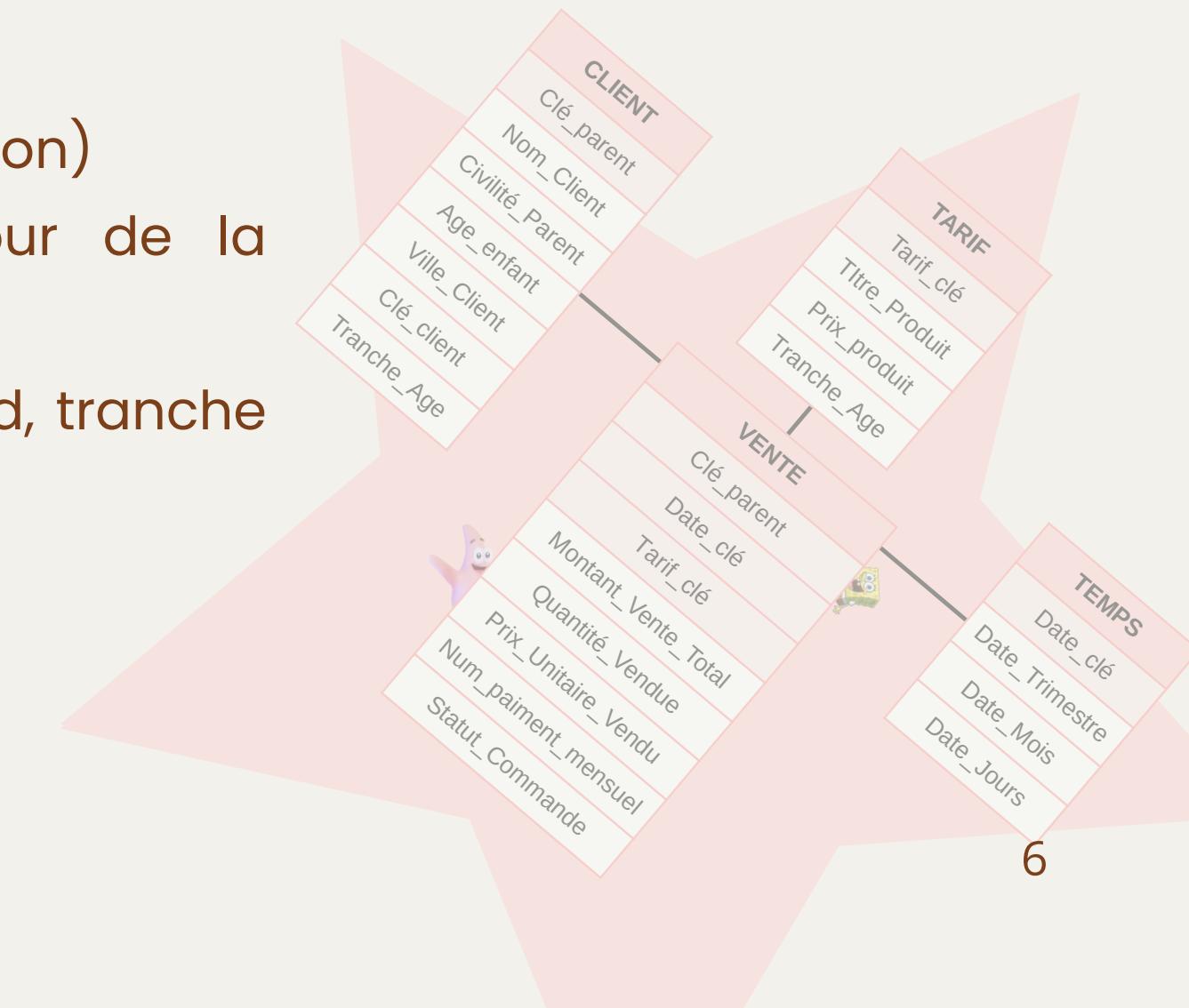


Table de Faits :

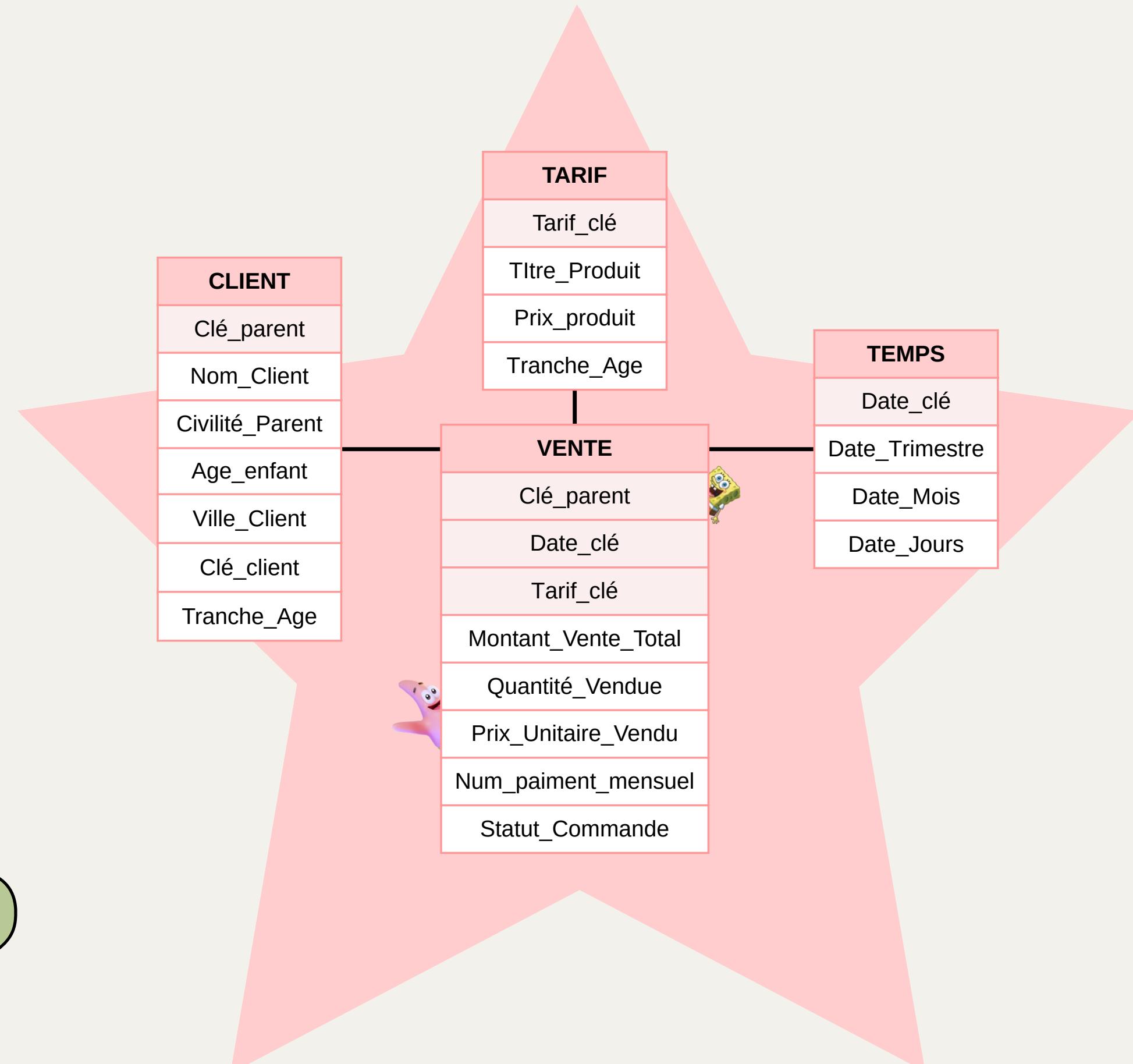
- Fait\_Vente (Contient les clés des dimensions et les mesures : Montant de la vente, Nombre d'unités vendues, Statut du paiement, etc...).

Tables de Dimensions :

- Dim\_Client (Qui achète ? Profil, âge, localisation)
- Dim\_Temps (Quand ? Mois, trimestre, jour de la semaine)
- Dim\_Titre (Quoi ? Type de titre, prix standard, tranche d'âge)



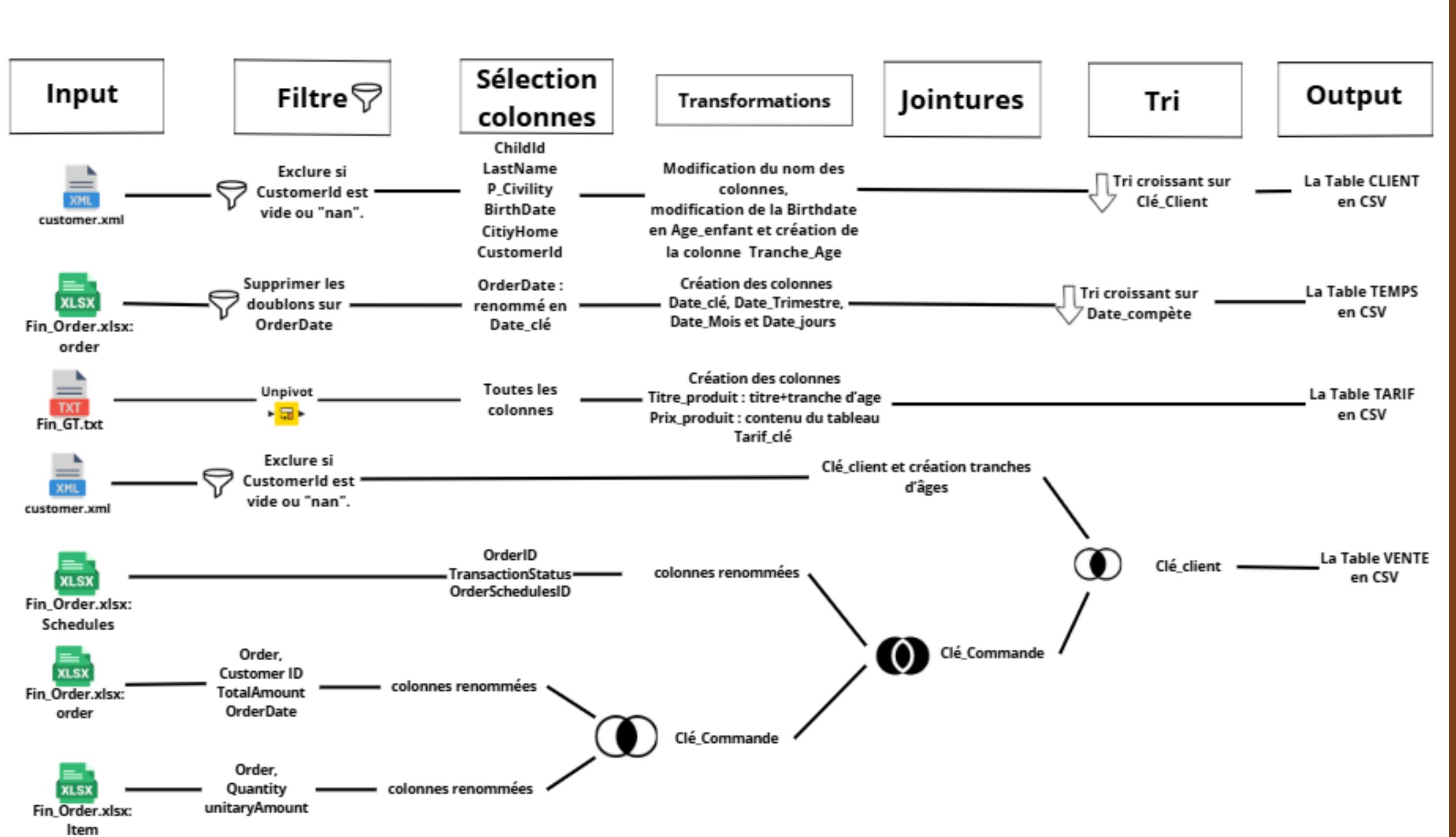
# MODÈLE EN ÉTOILE



2025

# ÉTAPE 3 : PLAN DE TRANSFORMATION DES DONNÉES

## PROCESSUS ETL

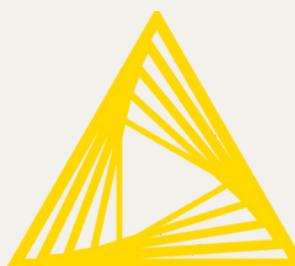


# ÉTAPE 4

# IMPLÉMENTATION

# MySQL ET KNIME

OUTIL  
ETL



Open for Innovation

**KNIME**

STRUCTURE  
BDD



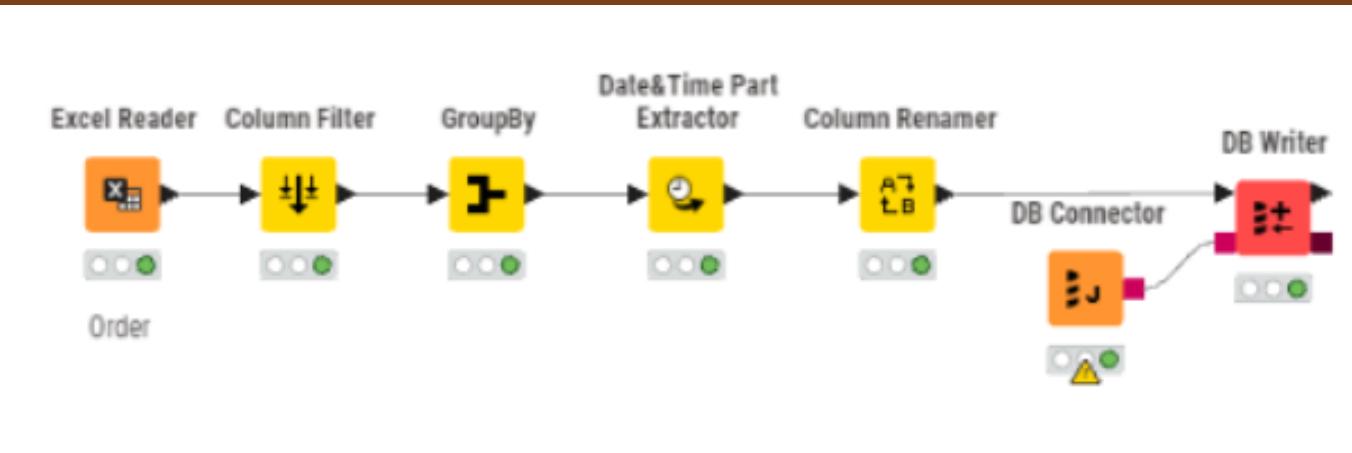
VOIR LA PROCHAINE PAGE



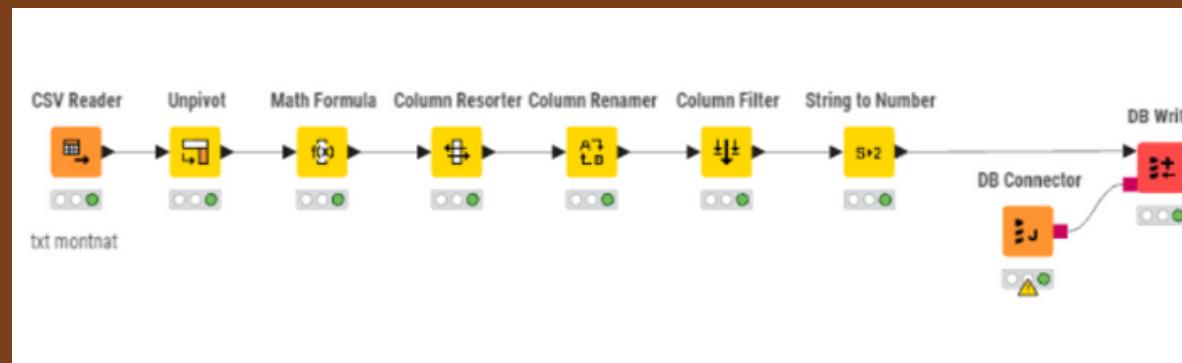
# FLUX KLIME



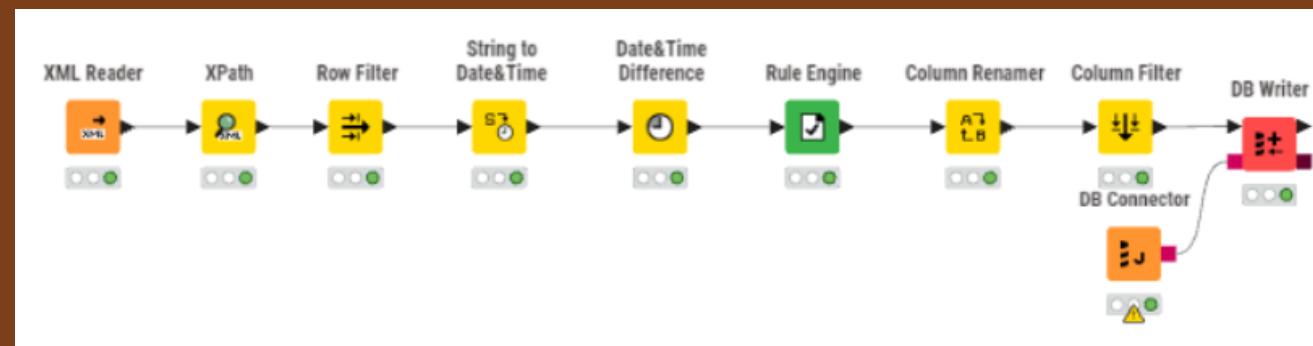
## DIMENSION TEMPS



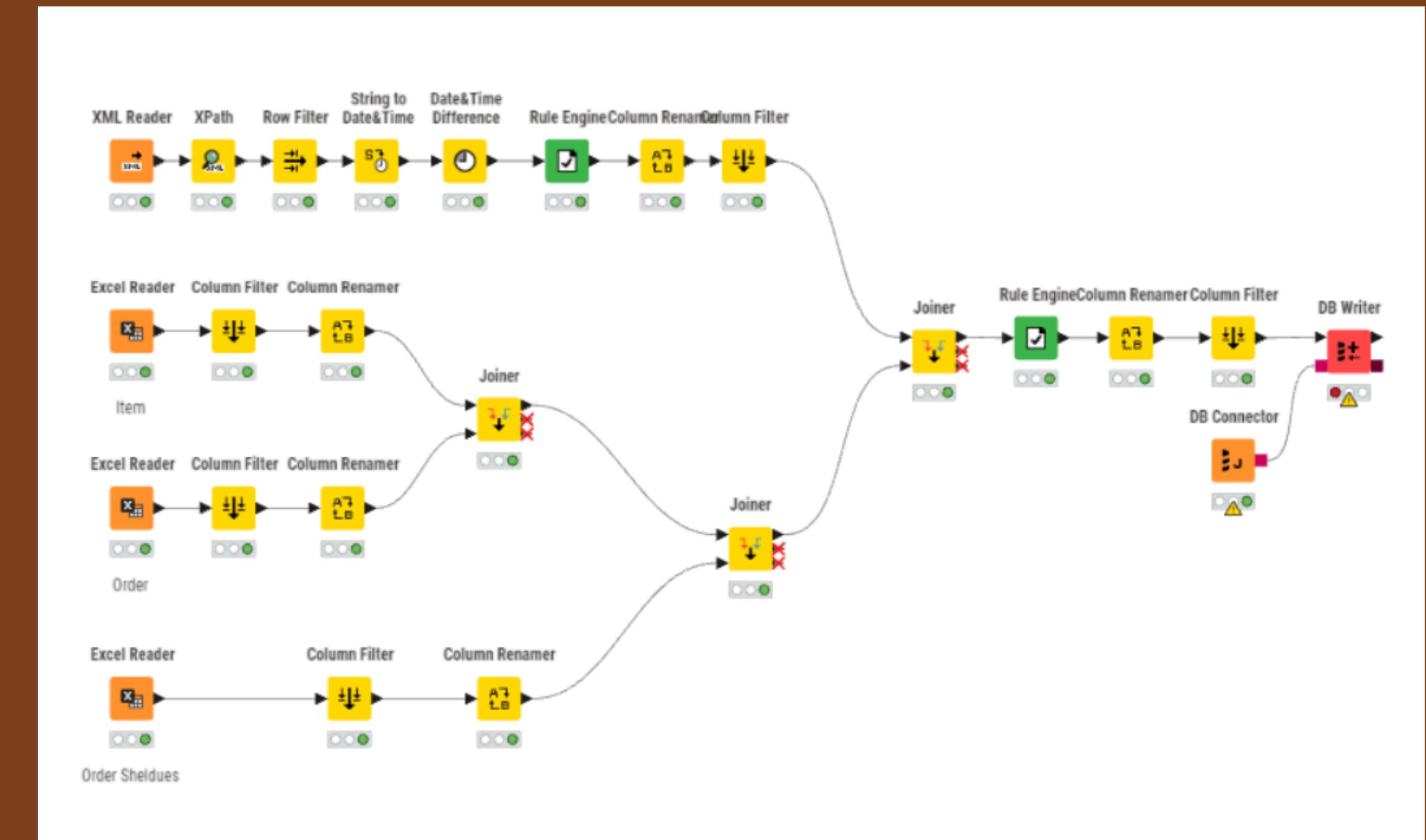
## DIMENSION TARIF



## DIMENSION CLIENT



## TABLE DE FAIT VENTE



# Démonstration

*“ON EFFACE TOUT ET  
ON RECOMMENCE”*

	Table	Action	Rows
<input type="checkbox"/>	Client		259
<input type="checkbox"/>	Tarif		12
<input type="checkbox"/>	Temps		58
<input type="checkbox"/>	vente		555

# RÉPONSE AUX PROBLéMATIQUES

• QUEL EST LE PROFIL DES PERSONNES PRéSENTES DANS CETTE BASE ?

```
SELECT
    Ville_Client,
    COUNT(Clé_Client) AS Nombre_Clients
FROM
    Client
WHERE
    Ville_Client IS NOT NULL
GROUP BY
    Ville_Client
ORDER BY
    Nombre_Clients DESC
LIMIT 5;
```

(TOP 5 DES VILLES Où RÉSIDENT  
LES CLIENTS)

```
SELECT
    'Répartition par âge' as Categorie,
    Tranche_Age as Valeur,
    COUNT(*) as Nombre,
    ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Client)), 2) as Pourcentage
FROM Client
GROUP BY Tranche_Age
UNION ALL
SELECT
    'Répartition par civilité parent' as Categorie,
    Civilité_Parent as Valeur,
    COUNT(*) as Nombre,
    ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Client)), 2) as Pourcentage
FROM Client
GROUP BY Civilité_Parent
ORDER BY Categorie, Nombre DESC;
```

Ville_Client	Nombre_Clients
ECHIROLLES	84
GRENOBLE	70
ST MARTIN D HERES	11
Fontaine	9
Saint-Martin-d'Hères	7

• QUEL EST LE TITRE LE PLUS POPULAIRE ?

```
SELECT
    t.Titre_Produit,
    COUNT(v.Clé_Parent) as Nombre_Ventes,
    SUM(v.Quantité_Vendue) as Quantité_Totale_Vendue
FROM vente v
INNER JOIN Tarif t ON v.Tarif_clé = t.Tarif_Clé
GROUP BY t.Titre_Produit
ORDER BY Nombre_Ventes DESC, Quantité_Totale_Vendue DESC;
```

Titre_Produit	Nombre_Ventes	Quantité_Totale_Vendue
Abo 30 Jou	105	944
10 voyages	73	568
1 voyage	8	493
Abo 1 an	368	368

L'ABONNEMENT DE 30 JOURS EST  
LE PLUS POPULAIRE !

## • QUELLES SONT LES VENTES MENSUELLES ?

Mois	CA_Mensuel	Nb_Commandes	Clients_Uniques
septembre	77327.50	465	169
octobre	12595.50	90	90

```

SELECT
    t.Date_Mois as Mois,
    ROUND(SUM(v.Montant_Vente_Total), 2) as CA_Mensuel,
    COUNT(v.Clé_Parent) as Nb_Commandes,
    COUNT(DISTINCT v.Clé_Parent) as Clients_Uniques
FROM vente v
INNER JOIN Temps t ON v.Date_Clé = t.Date_Clé
GROUP BY t.Date_Mois
ORDER BY MIN(v.Date_Clé);

```

## • QUI SONT LES PROFILS QUI ONT UN PAIEMENT 2 MOIS CONSÉCUTIF EN ERREUR ? 3 ?

Nom_Client	Age_Enfant	Ville_Client	Erreurs_Paiement	Total_Commandes
Givord	11	Grenoble	3	8
Torres Nava	12	ST MARTIN D HERES	3	8
Brandner	5	SEYSSINS	2	6

```

SELECT
    Client.Nom_Client,
    Client.Age_Enfant,
    Client.Ville_Client,
    COUNT(CASE WHEN vente.Statut_Commande = 'ERROR' THEN 1 END) as Erreurs_Paiement,
    COUNT(vente.Clé_Parent) as Total_Commandes
FROM Client
INNER JOIN vente ON Client.Clé_Parent = vente.Clé_Parent
GROUP BY Client.Clé_Parent, Client.Nom_Client, Client.Age_Enfant, Client.Ville_Client
HAVING Erreurs_Paiement >= 2
ORDER BY Erreurs_Paiement DESC;

```

# RÉPARTITION DU TRAVAIL

YANIS



- DICTIONNAIRE DES DONNées
- REQUÊTE SQL
- RENDUS ET RÉPONSES AUX PROBLèMATIQUES

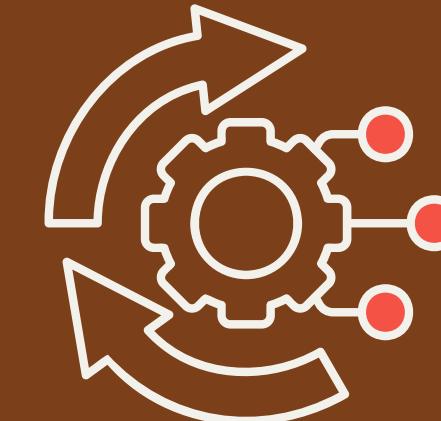


MySQL®

JONATHAN



- MODÈLE EN éTOILE
- PLAN DE TRANSFORMATION
- KNIME : TOUS LES FLUX DE TRANSFORMATIONS



Open for Innovation  
KNIME

GROUPE : JONATHAN WILD - YANIS STENTZEL

2025

# CONCLUSION



MERCI!

