

Schéma relationnel textuel (PostgreSQL) – Projet 5

Ce document décrit le schéma relationnel PostgreSQL mis en place dans le cadre de l'étape 4 du Projet 5 – Déploiement d'un modèle de Machine Learning.

Il vise à reconstruire en SQL les données utilisées dans le pipeline Python, à partir de trois sources CSV distinctes : SIRH, Évaluations annuelles, et Sondage bien-être.

Table **SIRH** (source brute RH)

Nom colonne	Type SQL	Contraintes
id_employee	INTEGER	PRIMARY KEY
age	INTEGER	NOT NULL
genre	VARCHAR(10)	NOT NULL
revenu_mensuel	INTEGER	NOT NULL
statut_marital	VARCHAR(20)	
departement	VARCHAR(50)	
poste	VARCHAR(50)	
nombre_experiences_precedentes	INTEGER	
nombre_heures_travailless	INTEGER	
annee_experience_totale	INTEGER	
annees_dans_l_entreprise	INTEGER	
annees_dans_le_poste_actuel	INTEGER	



Table **evaluations annuelles**

Nom colonne	Type SQL	Contraintes / Remarques
satisfaction_employee_environnement	INTEGER	
note_evaluation_precedente	INTEGER	
niveau_hierarchique_poste	INTEGER	
satisfaction_employee_nature_travail	INTEGER	
satisfaction_employee_equipe	INTEGER	
satisfaction_employee_equilibre_pro_perso	INTEGER	
eval_number	VARCHAR(10)	Contient id_employee avec préfixe E_ à nettoyer

Nom colonne	Type SQL	Contraintes / Remarques
note_evaluation_actuelle	INTEGER	
heure_supplementaires	VARCHAR(5)	Valeurs: 'Oui'/'Non'
augmentation_salaire_precedente	VARCHAR(10)	Valeurs: '12 %', '0 %' à transformer

🔗 **Important** : La colonne `eval_number` remplace `id_employee` mais nécessite un nettoyage (suppression du préfixe `E_`) et une conversion en `INTEGER`.
Ce champ est ensuite utilisé comme clé étrangère pour la jointure avec `sirh`.

📊 Table `sondage_bien_etre`

Nom colonne	Type SQL	Contraintes / Remarques
id_employee	INTEGER	Clé étrangère → référence à <code>sirh.id_employee</code>
a_quitte_l_entreprise	VARCHAR(3)	Valeurs : 'Oui' / 'Non' à encoder en binaire
nombre_participation_pee	INTEGER	
nb_formations_suivies	INTEGER	
nombre_employee_sous_responsabilite	INTEGER	
code_sondage	INTEGER	À renommer ou recoder pour correspondre à <code>id_employee</code> lors de la jointure
distance_domicile_travail	INTEGER	
niveau_education	INTEGER	Niveau ordinal (1 à 4)
domaine_etude	VARCHAR(50)	Exemples : 'Infra & Cloud', 'Transformation Digitale', 'Autre'
ayant_enfants	VARCHAR(1)	Valeurs : 'Y' / 'N' à encoder en binaire
frequence_deplacement	VARCHAR(20)	Valeurs : 'Occasionnel', 'Fréquent', etc. → prévoir encodage ordinal
annees_depuis_la_derniere_promotion	INTEGER	
annes_sous_responsable_actuel	INTEGER	

🔗 **Note** : La jointure avec `sirh` se fera via `code_sondage`, qui correspond à `id_employee`. Vérification de la correspondance exacte à faire au préalable.

🔗 Relations entre les tables

- `sirh.id_employee` → **clé primaire**
- `sondage_bien_etre.code_sondage` → à transformer pour correspondre à `sirh.id_employee`
- `evaluations_annuelles.eval_number` (à nettoyer → `id_employee`) → **clé étrangère vers sirh**

⚠ Attention : `eval_number` devra être nettoyé (`E_` supprimé) pour être joint avec `id_employee`.

Table `dataset_pipeline_final`

📁

Fichier source Python : `data/employees_net_refacto.csv`

📊

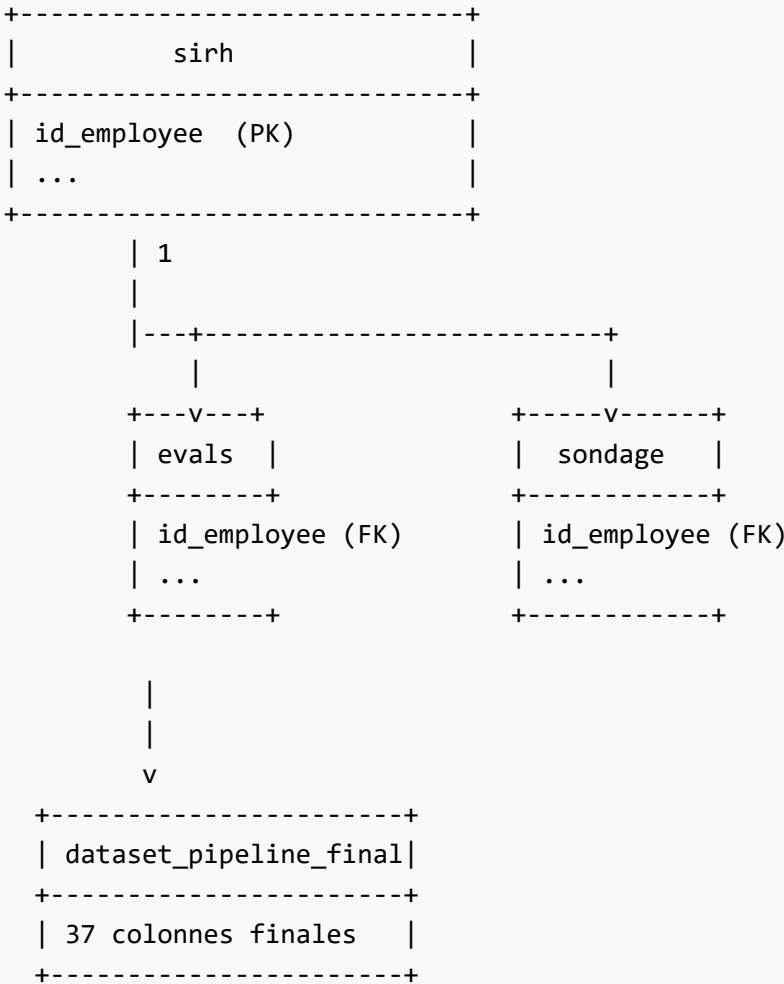
Colonnes : 37 variables, dont :

- Transformations numériques (`genre_binaire`, `age_revenu`, etc.)
- Encodages (`tranche_age`, `frequence_deplacement_num`)
- Variables dérivées (`interaction_satisfaction_anciennete`, etc.)

Cette table sera créée en SQL via une jointure des 3 sources, suivie des transformations reproduisant le pipeline Python.

Elle servira d'entrée pour le modèle de prédiction (équivalent du fichier `employees_net_refacto.csv`).

📐 Schéma relationnel simplifié (ASCII)



📊 Diagramme relationnel (format Mermaid)

SIRH ||--o{ EVALUATIONS_ANNUELLES : contient

```
SIRH ||--o{ SONDAGE_BIEN_ETRE : contient
SIRH ||--o{ DATASET_PIPELINE_FINAL : alimente

SIRH {
    INTEGER id_employee PK
    INTEGER age
    VARCHAR genre
    INTEGER revenu_mensuel
    VARCHAR statut_marital
    VARCHAR departement
    VARCHAR poste
    INTEGER nombre_experiences_precedentes
    INTEGER nombre_heures_travailless
    INTEGER annee_experience_totale
    INTEGER annees_dans_l_entreprise
    INTEGER annees_dans_le_poste_actuel
}

EVALUATIONS_ANNUELLES {
    VARCHAR eval_number
    INTEGER satisfaction_employee_environnement
    INTEGER note_evaluation_precedente
    INTEGER niveau_hierarchique_poste
    INTEGER satisfaction_employee_nature_travail
    INTEGER satisfaction_employee_equipe
    INTEGER satisfaction_employee_equilibre_pro_perso
    INTEGER note_evaluation_actuelle
    VARCHAR heure_supplementaires
    VARCHAR augmentation_salaire_precedente
}

SONDAGE_BIEN_ETRE {
    INTEGER code_sondage
    VARCHAR a_quitte_l_entreprise
    INTEGER nombre_participation_pee
    INTEGER nb_formationen_suivies
    INTEGER nombre_employee_sous_responsabilite
    INTEGER distance_domicile_travail
    INTEGER niveau_education
    VARCHAR domaine_etude
    VARCHAR ayant_enfants
    VARCHAR frequence_deplacement
    INTEGER annees_depuis_la_derniere_promotion
    INTEGER annes_sous_responsable_actuel
}

DATASET_PIPELINE_FINAL {
    INTEGER id_employee
    ... 37 colonnes finales ...
}
```

Liens externes

