Modélisation du Datawarehouse et Processus ETL
TP_ETL

Introduction abrégée

Les données sources proviennent d'une base opérationnelle (`Ventes`, `Clients`) et d'un fichier `prestations.csv`. L'objectif du datawarehouse est de permettre des analyses OLAP (prix moyens, durées d'intervention, comparaison lieu d'intervention vs résidence client, etc.). La granularité minimale géographique retenue est la **ville**. Nous nous concentrons sur la modélisation (Q1) puis le schéma relationnel (Q2).

1 Modélisation en étoile (Question 1)

Grain du fait

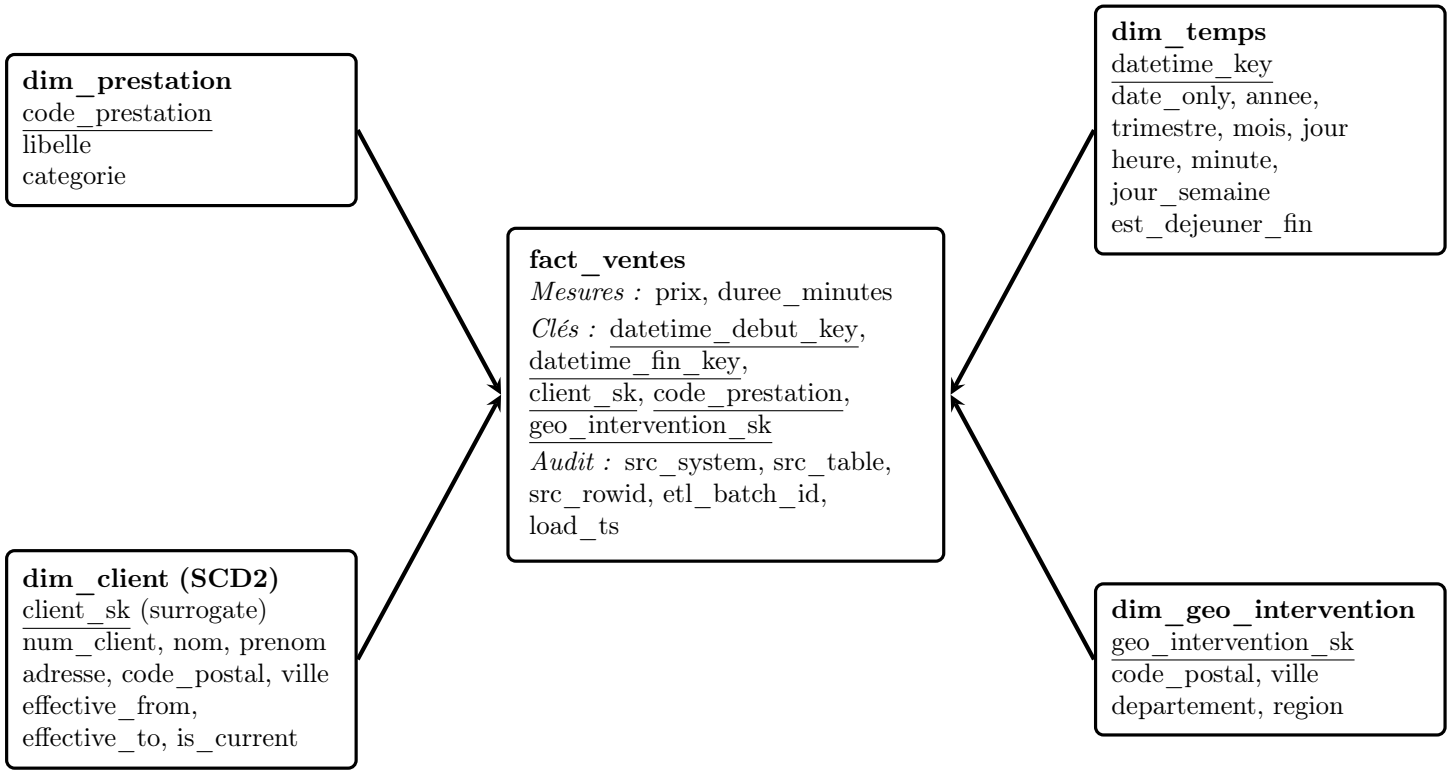
Une ligne du fait correspond à **une intervention/vente**. Mesures: `prix`, `duree_minutes`.

Dimensions

- **Temps** (rôle-jouée): une unique dimension `dim_temps` référencée deux fois par le fait (`datetime_debut_key`, `datetime_fin_key`). La clé est la chaîne datetime (cf. indication du sujet). Attributs dérivés: date, année, trimestre, mois, jour, heure, minute, jour de semaine, indicateur *fin sur l'heure du déjeuner*.
- **Prestation**: `code_prestation` (clé naturelle), libellé, catégorie (*Dépannage*/*Installation*).
- **Géographie d'intervention**: ville, code postal, département, région.
- **Client** (SCD2): `num_client` (clé naturelle), nom, prénom, adresse, code postal, ville, (département, région), période de validité (`effective_from`/`effective_to`), drapeau `is_current`.

Justification brève. *Grain strict* = 1 intervention garantit des mesures additives (prix, durée). *Temps rôle-jouée* (début/fin) évite de dupliquer la dimension et couvre les analyses « fin sur l'heure du déjeuner ». La *géographie d'intervention* séparée répond aux agrégations ville/département/région. *Client en SCD2* rend le DW résilient aux changements d'adresse: le fait pointe vers la version courante au moment de la vente.

Schéma conceptuel (étoile)



Remarque. Conformément à l'indication du sujet, nous **répétons les attributs géographiques** pour le lieu d'intervention (dimension dédiée) et pour la résidence du client (stockés dans `dim_client`).

2 Schéma relationnel et résilience/traçabilité (Question 2)

Clés et résilience

- **Dates/heures**: la clé des dates est la chaîne datetime `yyyy-MM-dd HH:mm` (`dim_temps.datetime_key`), garantissant l'unicité et alignée sur la source.
- **Client SCD2**: historique d'adresse via `effective_from`, `effective_to`, `is_current`; jointure du fait via *surrogate key* `client_sk`.

Schéma relationnel (résumé explicite)

- **dim_temps**(datetime_key, date_only, annee, trimestre, mois, jour, heure, minute, jour_semaine, est_weekend, est_dejeuner_fin)
- **dim_prestation**(code_prestation, libelle, categorie)
- **dim_geo_intervention**(geo_intervention_sk, code_postal, ville, departement, region), *UNIQUE*(code_postal, ville)
- **dim_client**(client_sk, num_client, nom, prenom, adresse, code_postal, ville, departement, region, effective_from, effective_to, is_current), *UNIQUE*(num_client, effective_from)
- **fact_ventes**(vente_sk, client_sk→dim_client, code_prestation→dim_prestation, geo_intervention_sk datetime_debut_key→dim_temps, datetime_fin_key→dim_temps, prix, duree_minutes, src_system, src_table, src_rowid, etl_batch_id, load_ts)

Traçabilité

Le fait stocke `src_*` (système/table/rowid), `etl_batch_id` et `load_ts`. Les rejets de qualité (dates incohérentes, FK non résolues, prix négatif) peuvent être conservés en fichiers CSV et/ou dans une table de rejets séparée.

Contraintes et qualité

`prix >= 0`, `duree_minutes >= 0`, `datetime_fin >= datetime_debut`. La cohérence temporelle est vérifiée côté ETL, puis la durée est calculée et stockée.