



# Documentation – Mapping Classification Produit (Excel → Modèle)



## Contexte

Le fichier Excel fournit les codes et libellés des familles marketing de produits ainsi que leur hiérarchie (Grande famille → Regroupement → Famille).

Ces données alimentent le modèle de données Sujet Produit, principalement la dimension `dim_produit_classification`, la dimension `dim_produit_classification_type`, et le bridge `bridge_classification_produit_hierarchie`.

---

## Tableau de Mapping

Colonne Excel (source)	Attribut Modèle (cible)	Table cible	Commentaire
NUM_GRAN_FAMI_MKTG_PRDI	id_cod_classification_produit (niveau 1)	dim_produit_classification (SCD1)	Identifiant de la grande famille marketing
DSC_GRAN_FAMI_MKTG_PRDI	nom_classification_produit (niveau 1)	dim_produit_classification (SCD1)	Nom de la grande famille marketing
TYP_REGT_MKTG_PRDI	desc_classification_produit	dim_produit_classification (SCD1)	Peut être repris comme description plus riche
NUM_REGT_MKTG_PRDI	id_cod_classification_produit (niveau 2)	dim_produit_classification (SCD1)	Identifiant du regroupement marketing
DSC_REGT_MKTG_PRDI	nom_classification_produit (niveau 2)	dim_produit_classification (SCD1)	Nom du regroupement marketing
TYP_REGT_MKTG_PRDI	desc_classification_produit	dim_produit_classification (SCD1)	Peut compléter la description
NUM_FAMI_MKTG_PRDI	id_cod_classification_produit (niveau 3)	dim_produit_classification (SCD1)	Identifiant de la famille marketing
DSC_FAMI_MKTG_PRDI	nom_classification_produit (niveau 3)	dim_produit_classification (SCD1)	Nom de la famille marketing détaillée
TYP_REGT_MKTG_PRDI	desc_type_classification_produit	dim_produit_classification_type (SCD1)	Type du regroupement (permet de catégoriser la classification)

## Gestion de la hiérarchie

- Les niveaux (1,2,3) sont stockés dans la même table `dim_produit_classification`, différenciés par l'attribut `num_niveau`.
  - Les relations parent → enfant sont modélisées dans `bridge_classification_produit_hierarchie` :
    - L1 → L2 : Grande famille → Regroupement
    - L2 → L3 : Regroupement → Famille
- 

## Gestion de l'unicité

- Chaque code (`NUM_GRAN`, `NUM_REGT`, `NUM_FAMI`) est unique dans son niveau.

Contrainte définie :

`UNIQUE (id_cod_classification_produit, num_niveau)`

- - Une clé technique `id_classification_sk` est utilisée comme PK.
-

## Exemple d'alimentation

Excel :

- `NUM_GRAN_FAMI_MKTG_PRDI = 2` → "Financement et cartes de crédit"
- `NUM_REGT_MKTG_PRDI = 4` → "Financement court terme"
- `NUM_FAMI_MKTG_PRDI = 83` → "Marge de crédit agricole"
- `TYP_REGT_MKTG_PRDI = "E"`

Chargement :

- `dim_produit_classification`
    - (2, "Financement et cartes de crédit", niveau 1)
    - (4, "Financement court terme", niveau 2)
    - (83, "Marge de crédit agricole", niveau 3)
  - `dim_produit_classification_type`
    - "E"
  - `bridge_classification_produit_hierarchie`
    - (parent=2, enfant=4)
    - (parent=4, enfant=83)
-

👉 Cette structure permet :

- une gestion centralisée des niveaux de classification,
- une évolutivité (ajout d'un niveau 4 possible sans recréer une table),
- un contrôle strict via les contraintes d'unicité et le bridge parent-enfant.

Mapping avec ton modèle de données Sujet Produit :

1. dim\_produit\_classification (SCD1)

- **nom\_classification\_produit** = DSC\_FAMI\_MKTG\_PRDI (nom famille marketing détaillée, ex: "Carte de crédit", "Epargne stable")
- **desc\_classification\_produit** = combinaison de DSC\_REGT\_MKTG\_PRDI et DSC\_GRAN\_FAMI\_MKTG\_PRDI (description du regroupement et de la grande famille)
- **num\_niveau** = dérivé de  $\frac{\text{NUM\_GRAN\_FAMI\_MKTG\_PRDI}}{\text{NUM\_REGT\_MKTG\_PRDI} \times \text{NUM\_FAMI\_MKTG\_PRDI}}$  (hiérarchie)

2. dim\_produit\_classification\_type (SCD1)

- **desc\_type\_classification\_produit** = TYP\_REGT\_MKTG\_PRDI (ex: "E" pour épargne,

"F" pour financement, etc.)

### 3. bridge\_classification\_produit\_hierarchie

- **id\_cod\_classification\_parent** = valeurs des colonnes NUM\_GRAN\_FAMI\_MKTG\_PRDI ou NUM\_REGT\_MKTG\_PRDI selon le niveau
- **id\_cod\_classification\_enfant** = valeurs de NUM\_REGT\_MKTG\_PRDI ou NUM\_FAMI\_MKTG\_PRDI  
👉 Ce bridge va reconstituer la hiérarchie Grande Famille → Regroupement → Famille marketing.

## Hiérarchie à alimenter (Bridge)

- bridge\_classification\_produit\_hierarchie
  - **id\_cod\_classification\_parent** = NUM\_GRAN\_FAMI\_MKTG\_PRDI (ou NUM\_REGT si on descend au niveau suivant)
  - **id\_cod\_classification\_enfant** = NUM\_REGT\_MKTG\_PRDI ou NUM\_FAMI\_MKTG\_PRDI
  - **dat\_effective** / **dat\_expiree** = à gérer via la date de référence de ton fichier

Excel :

- NUM\_GRAN\_FAMI\_MKTG\_PRDI = 2
- DSC\_GRAN\_FAMI\_MKTG\_PRDI = "Financement et cartes de crédit"
- NUM\_REGT\_MKTG\_PRDI = 4
- DSC\_REGT\_MKTG\_PRDI = "Financement court terme"

- NUM\_FAMI\_MKTG\_PRDI = 83
- DSC\_FAMI\_MKTG\_PRDI = "Marge de crédit agricole"
- TYP\_REGT\_MKTG\_PRDI = "E"

#### 👉 Mapping :

- dim\_produit\_classification :
  - id\_cod\_classification\_produit = 83
  - nom\_classification\_produit = "Marge de crédit agricole"
  - desc\_classification\_produit = "Financement court terme / Financement et cartes de crédit"
  - num\_niveau = 3
- dim\_produit\_classification\_type :
  - desc\_type\_classification\_produit = "E"
- bridge\_classification\_produit\_hierarchie :
  - parent = 4 (Financement court terme)
  - enfant = 83 (Marge de crédit agricole)

un modèle "multi-niveau dans une seule dimension".

Concrètement :

- La même table **dim\_produit\_classification** contient les lignes de tous les niveaux (L1 = Grande famille, L2 = Regroupement, L3 = Famille).
- Les mêmes attributs (ex: **id\_cod\_classification\_produit**, **nom\_classification\_produit**) sont donc répétés pour chaque niveau, et c'est la colonne **num\_niveau** qui dit à quel niveau appartient la ligne (1, 2 ou 3).
- La hiérarchie entre niveaux n'est pas portée par la clé elle-même, elle est portée par le bridge **bridge\_classification\_produit\_hierarchie** qui relie *parent* ↔ *enfant*.

## Pourquoi ça marche

- Les attributs “code” et “nom” existent à chaque niveau (tu as bien un code et un libellé pour L1, L2, L3 dans ton Excel).
- Les règles métier sont partagées: un code, un nom, des dates, un type, etc. → un schéma unique évite 3 tables quasi identiques.
- Le `num_niveau` permet de filtrer/contrôler (ex: `WHERE num_niveau = 3` pour récupérer les familles “feuilles”).

## Ce qu’il faut ajouter pour être solide





1. Clé technique (surrogate key): `id_classif_sk` (PK).
2. Unicité: contrainte UNIQUE sur (`source_system`, `id_cod_classification_produit`, `num_niveau`) pour éviter qu’un même code existe à 2 niveaux.
3. Bridge parent-enfant:
  - `L1→L2` (parent = code L1, enfant = code L2)
  - `L2→L3` (parent = code L2, enfant = code L3)

## Chargement (ETL) en 3 étapes

1. L1: insérer `NUM_GRAN...`, `DSC_GRAN...` avec `num_niveau=1`.
2. L2: insérer `NUM_REGT...`, `DSC_REGT...` avec `num_niveau=2`.
3. L3: insérer `NUM_FAMI...`, `DSC_FAMI...` avec `num_niveau=3`.
4. Bridge: construire les paires (`L1→L2`, `L2→L3`) à partir des colonnes de la même ligne Excel.

## Avantages vs. alternatives



-  **Une seule dimension:** plus simple à joindre, même schéma pour tous les niveaux.
-  **Évolutif:** si demain tu ajoutes un niveau 4, tu n'ajoutes qu'une valeur de `num_niveau`.
-  **Discipline de données:** il faut la contrainte d'unicité + le bridge pour ne pas mélanger les niveaux.
-  **Alternative:** 3 tables séparées (`dim_L1`, `dim_L2`, `dim_L3`) ou une dimension auto-référencée (`parent_id`). C'est plus verbeux, offre un peu plus de contrôle, mais alourdit le modèle.

En bref: les mêmes attributs peuvent stocker des niveaux différents parce que le niveau est explicitement porté par `num_niveau`, et la relation hiérarchique est externalisée dans le bridge. C'est un pattern classique de "classification multi-niveau" et il colle parfaitement à ton Excel.

**L'unicité**, ici, c'est la règle qui garantit qu'on n'a pas deux lignes différentes représentant la *même classification au même niveau*.

Dans ton modèle, comme tous les niveaux (1,2,3) sont stockés dans la même table `dim_produit_classification`, il faut éviter par exemple :

- d'avoir **deux fois le même code** `NUM_FAMI_MKTG_PRDI = 83` (ex: "Marge de crédit agricole")
- ou d'avoir une "grande famille" dupliquée avec le même code mais orthographiée différemment.

## Comment définir cette unicité ?

- **Clé technique (PK)** : un identifiant interne `id_classification_sk` (clé substitut).

**Contrainte d'unicité** sur :

`(id_cod_classification_produit, num_niveau)`

- → ça signifie : *un même code ne peut exister qu'une seule fois à un niveau donné.*

## Exemple

- Niveau 1 : `NUM_GRAN_FAMI_MKTG_PRDI = 2`, "Financement et cartes de crédit"
- Niveau 2 : `NUM_REGT_MKTG_PRDI = 4`, "Financement court terme"
- Niveau 3 : `NUM_FAMI_MKTG_PRDI = 83`, "Marge de crédit agricole"

On peut donc avoir **83 uniquement au niveau 3**, pas au niveau 2 ou 1.  
L'unicité empêche de réutiliser le même code dans plusieurs contextes.

## Pourquoi c'est important ?

- Ça évite les doublons dans tes dimensions.
- Ça garantit la cohérence de ton **bridge parent-enfant**, car chaque enfant pointe vers un parent unique

Voici un exemple de **DDL SQL** pour ta table `dim_produit_classification` avec :

### 1. Clé primaire technique (surrogate key)

### 2. Contrainte d'unicité (`id_cod_classification_produit + num_niveau`)

```
CREATE TABLE dim_produit_classification (
  id_classification_sk BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY, --
  clé technique interne
  id_cod_classification_produit VARCHAR(50) NOT NULL, -- code issu Excel
  (NUM_GRAN, NUM_REGT, NUM_FAMI)
  id_produit_classification_type VARCHAR(10), -- type (TYP_REGT_MKTG_PRDI)
  nom_classification_produit VARCHAR(255) NOT NULL, -- libellé (DSC_*)
  desc_classification_produit VARCHAR(500), -- description enrichie
  num_niveau SMALLINT NOT NULL, -- niveau hiérarchique (1,2,3)
  dat_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  dat_modification TIMESTAMP,

  CONSTRAINT uq_dim_produit_classification
  UNIQUE (id_cod_classification_produit, num_niveau) -- 🔒 unicité du code par niveau
);
```

## Explication

- **id\_classification\_sk** : identifiant technique interne (jamais exposé, sert de PK dans ton DWH).
- **id\_cod\_classification\_produit** : code métier venant d'Excel (NUM\_GRAN, NUM\_REGT, NUM\_FAMI).
- **num\_niveau** : indique si c'est une **grande famille (1)**, un **regroupement (2)** ou une **famille détaillée (3)**.
- **Contrainte UNIQUE** : empêche que le même code apparaisse deux fois dans le même niveau

### Exemple d'insertion valide

```
INSERT INTO dim_produit_classification
(id_cod_classification_produit, nom_classification_produit,
num_niveau)
VALUES ('2', 'Financement et cartes de crédit', 1);
```

```
INSERT INTO dim_produit_classification
(id_cod_classification_produit, nom_classification_produit,
num_niveau)
VALUES ('4', 'Financement court terme', 2);
```

```
INSERT INTO dim_produit_classification
(id_cod_classification_produit, nom_classification_produit,
num_niveau)
VALUES ('83', 'Marge de crédit agricole', 3);
```

### Exemple rejeté

```
INSERT INTO dim_produit_classification
(id_cod_classification_produit, nom_classification_produit,
num_niveau)
VALUES ('83', 'Marge agricole duplicat', 3);
```

 rejeté car (83, niveau=3) existe déjà.