

SAE S1.04(R1.05) Groupe 3:

Aldo Alex NGANJI

Aminata Sita CAMARA

S1 - S1.04 Création BD

R1.05 Introduction BD

Partie 1 : Correction du Schéma Existant

Avant d'ajouter de nouvelles fonctionnalités, notre priorité était de nettoyer et de normaliser la base de données existante. Nous avons identifié deux zones principales où la conception originale manquait de clarté et n'était pas évolutive ("scalable") :

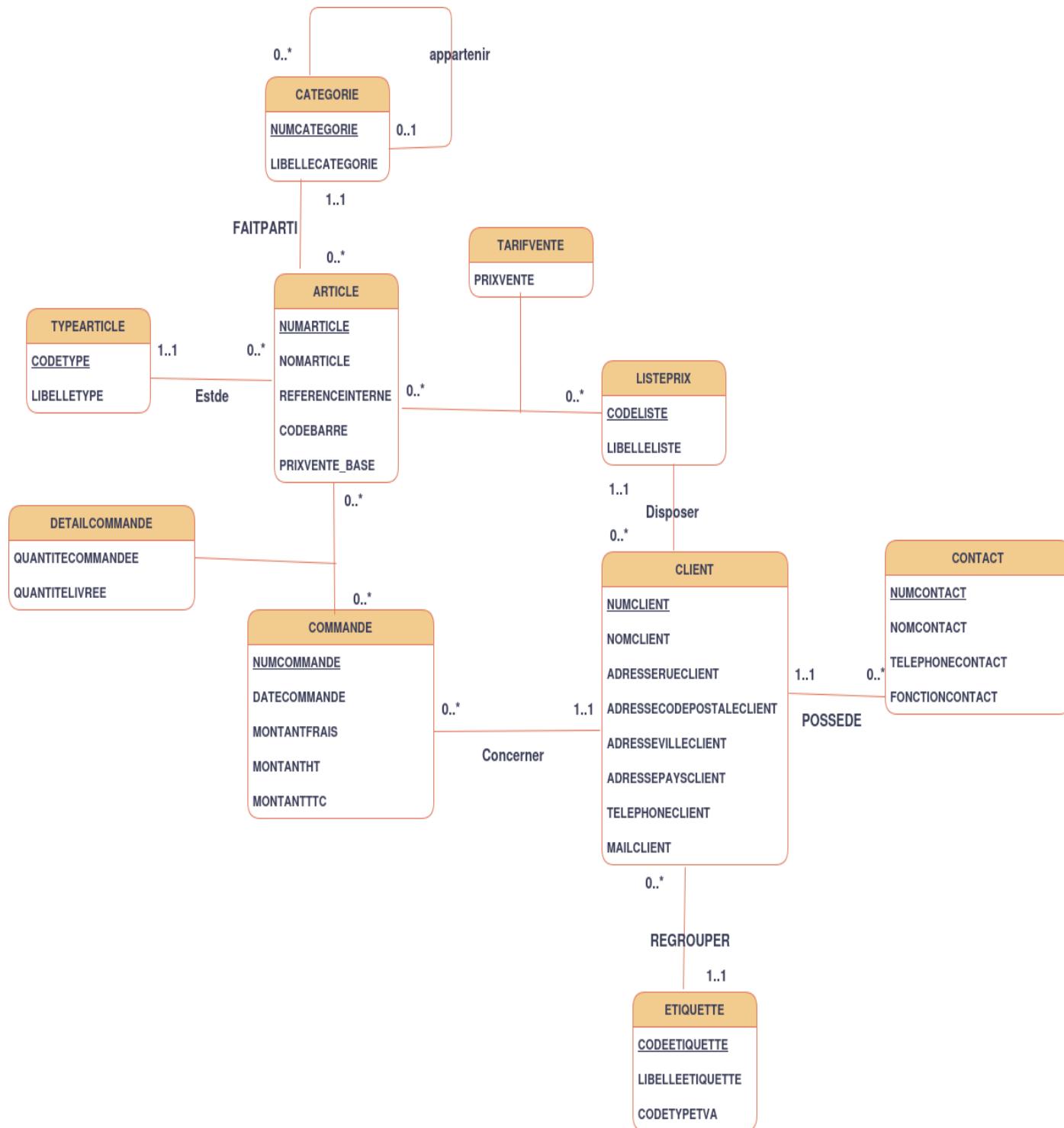
1. La Faille des Contacts Client (Dénormalisation)

- **Le Problème :** La table **CLIENT** était rigide. Elle utilisait des colonnes fixes comme **NOMCONTACT1**, **TELEPHONECONTACT1**, **NOMCONTACT2**, ...
- **Pourquoi c'est une faille :** Que se passe-t-il si un client a trois contacts ? Ou zéro ? Cette conception n'est pas flexible, gaspille de l'espace.
- **Notre Solution :** Nous avons créé une nouvelle table **CONTACT** séparée. Cela permet à un client d'avoir *n'importe quel nombre* de contacts (zéro, un ou vingt) de manière propre et efficace.

2. La Faille de l'Ambiguïté des Prix

- **Le Problème :** Le schéma était confus. **PRIXVENTE** (prix de vente) existait dans la table **ARTICLE** et aussi dans **TARIFVENTE**, rendant la logique difficile à suivre. De plus, **COUTACHAT** (coût d'achat) était stocké dans **ARTICLE**.
- **Pourquoi c'est une faille :** Le **COUTACHAT** n'appartient pas à l'article ; il dépend du *fournisseur*. Et les deux colonnes **PRIXVENTE** rendaient la conception ambiguë pour les futurs développeurs.
- **Notre Solution :**
 1. Nous avons clarifié la logique en renommant **ARTICLE.PRIXVENTE** en **PRIXVENTE_BASE**. Cela indique clairement qu'il s'agit du prix *par défaut*.
 2. Nous avons **supprimé COUTACHAT** de la table **ARTICLE**. Cet attribut est logiquement lié au fournisseur qui vend l'article. Puisque la table **FOURNISSEUR** n'existe pas dans cette "Partie 1", la présence de **COUTACHAT** n'était pas nécessaire. Il sera réintroduit de manière correcte lorsque nous ajouterons la gestion des fournisseurs pour la partie 2.

Le diagramme de classes modifié



le schéma de données des modifications

- CLIENT (NUMCLIENT, NOMCLIENT, ADRESSERUECLIENT, ADRESSECODEPOSTALECLIENT, ADRESSEVILLECLIENT, ADRESSEPATSCCLIENT, TELEPHONECLIENT, MAILCLIENT, #CODEETIQUETTE, #CODELISTE)
- CONTACT(NUMCONTACT, NOMCONTACT, TELEPHONECONTACT, FONCTIONCONTACT, #NUMCLIENT)
- ARTICLE(NUMARTICLE, NOMARTICLE, REFERENCEINTERNE, CODEBARRE, PRIXVENTE_BASE, #NUMCATEGORIE, #CODETYPE)

le script SQL des modifications

1. Suppression des colonnes dénormalisées de la table CLIENT

```
ALTER TABLE CLIENT
DROP COLUMN NOMCONTACT1,
DROP COLUMN TELEPHONECONTACT1,
DROP COLUMN FONCTIONCONTACT1,
DROP COLUMN NOMCONTACT2,
DROP COLUMN TELEPHONECONTACT2,
DROP COLUMN FONCTIONCONTACT2;
```

2. Création de la nouvelle table normalisée CONTACT

```
CREATE TABLE CONTACT (
    NUMCONTACT SERIAL PRIMARY KEY,
    NOMCONTACT VARCHAR(50),
    TELEPHONECONTACT VARCHAR(12),
    FONCTIONCONTACT VARCHAR(20),
    NUMCLIENT INTEGER NOT NULL,
    FOREIGN KEY (NUMCLIENT) REFERENCES CLIENT(NUMCLIENT) ON DELETE CASCADE
);
```

3. Renommage de PRIXVENTE en PRIXVENTE_BASE pour clarifier son rôle

```
ALTER TABLE ARTICLE
RENAME COLUMN PRIXVENTE TO PRIXVENTE_BASE;
```

4. Suppression de COUTACHAT (en attente de la table FOURNISSEUR)

```
ALTER TABLE ARTICLE
DROP COLUMN COUTACHAT;
```