

Passage MCD <=> modèle relationnel

I. Rappel sur le modèle relationnel

C'est un modèle LOGIQUE de donnée, celui qui correspond à l'organisation des données dans les bases de données relationnelles (il existe d'autres organisations de bases de données : hiérarchique, réseau, objet,...).

Les SGBD actuels les plus courants sont relationnels (Oracle, SQL Server, Access, MySql, ...)

Un modèle relationnel est composé de **relations**, encore appelée **tables**.

Ces tables sont décrites par des **attributs** ou **champs** (noms de colonnes). Pour décrire une relation, on indique tout simplement son nom en majuscule, suivi du nom de ses attributs entre parenthèses.

L'identifiant d'une relation est composé d'un ou plusieurs attributs qui forment la **clé primaire**. Une relation peut faire référence à une autre en utilisant une **clé étrangère**, qui correspond à la clé primaire de la relation référencée.

Il n'y a **pas de notation officielle** pour repérer les clés primaires et étrangères. C'est à vous d'en adopter une et de l'expliquer en légende.

Toutefois, une notation s'est peu à peu répandue en BTS IG (celle que nous utiliserons):

- on souligne la clé primaire d'un seul trait
- on fait précéder (ou suivre) les clés étrangères du symbole #

Chaque ligne (**tuple** ou **enregistrement**) d'une table représente une **occurrence** de l'entité ou de l'association correspondante.

II. Passage du MCD au MR

A. Règle 1

Toute entité devient une relation ayant pour clé primaire son identifiant. Chaque propriété se transforme en attribut.

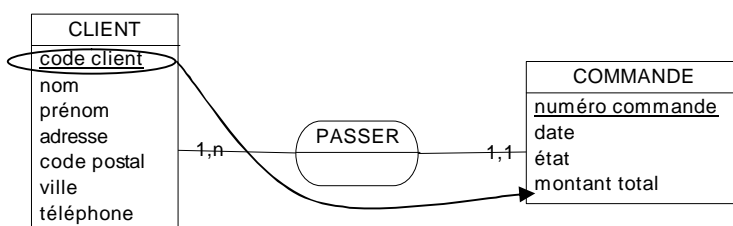
CLIENT
<u>code client</u>
nom
prénom
adresse
code postal
ville
téléphone

==> CLIENT (code_client, nom, prénom, adresse, code_postal, ville, téléphone)

Remarque : contrairement aux propriétés, les attributs ne doivent pas comporter d'espaces.

B. Règle 2

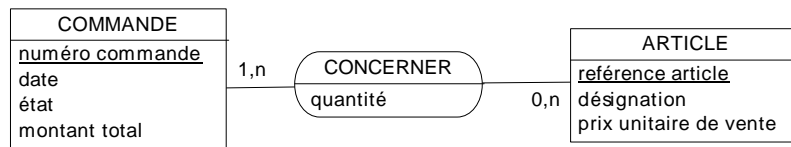
Toute association hiérarchique (de type [1, n]) se traduit par une clé étrangère. La clé primaire correspondant à l'entité père (côté n) migre comme clé étrangère dans la relation correspondant à l'entité fils (côté 1).



COMMANDE(numéro_commande, date, état, montant_total, # code_client)

C. Règle 3

Toute association non hiérarchique (de type $[n, n]$ ou de dimension > 2) **devient une relation**. La clé primaire est formée par la concaténation (juxtaposition) l'ensemble des identifiants des entités reliées. Toutes les propriétés éventuelles deviennent des attributs qui ne peuvent pas faire partie de la clé.

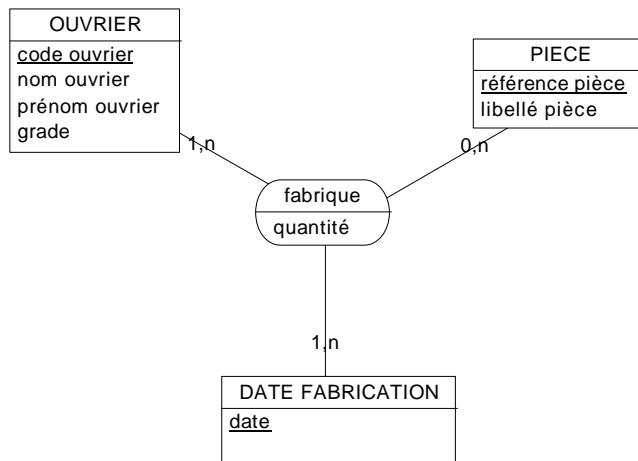


CONCERNER(#numéro_commande, #référence_article, quantité)

Cette règle est valable pour toutes les associations ternaires (ou quaternaires) qui sont forcément non hiérarchiques (cardinalités maximales toutes égales à n).

D. Exception à la règle 1

Les entités n'ayant que leur identifiant comme attribut ne deviennent pas des relations, mais des attributs dans les autres relations liées.



Avec ce modèle, on mémorise chaque jour pour chaque ouvrier les pièces qu'il a fabriqué et en quelle quantité.

Quand on passe au modèle relationnel, l'entité **DATE FABRICATION** ne devient pas une relation, mais un attribut clé dans la relation **FABRIQUE** issue de l'association.

~~DATE FABRICATION(date)~~ fait partie de la clé primaire, mais n'est pas clé étrangère

FABRIQUE(#code_ouvrier, #référence_pièce, date, quantité).

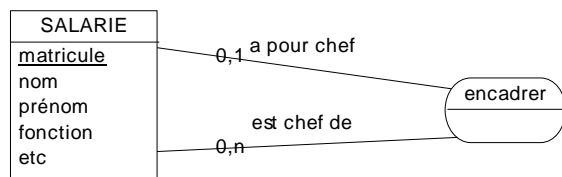
E. Cas particulier des associations réflexives

Les **associations réflexives** suivent les règles 2 ou 3 selon les cardinalités mais posent un problème particulier : une même propriété va se retrouver deux fois en attribut dans la même relation. Il faut alors donner un nom différent et significatif aux deux attributs correspondants.

Dans les réflexives, il est conseillé de nommer les branches par des **rôles** pour pouvoir lire dans le bon sens l'association. Les rôles aident à nommer les attributs correspondant à l'association.

Réflexive hiérarchique (une branche à la cardinalité maxi à 1 et l'autre à n)

✍ règle n° 2



Lecture de l'association :

Un salarié a pour chef 0 ou un seul autre salarié. Un salarié est chef de 0 à n autre(s) salarié.

Règle n°1: l'identifiant de SALARIE va devenir clé primaire et les autres propriétés des attributs

Règle n°2 : pour traduire l'association [1, n] encadrer, l'identifiant de l'entité SALARIE devient clé étrangère

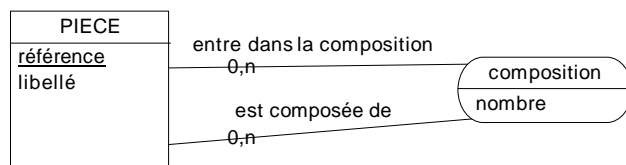
✍ l'identifiant de SALARIE matricule se retrouve deux fois dans la relation : comme clé primaire et comme clé étrangère

On va donc donner un nom différent et significatif à ces deux matricules, par exemple

Traduction en modèle relationnel

SALARIE(matricule, nom, prénom, fonction,..., #matricule_chef)**Réflexive non hiérarchique**

✍ règle n°3



Lecture de l'association

Une pièce entre dans la composition de 0 à plusieurs autres pièces. Une pièce peut être composée de plusieurs autres pièces. Une pièce entre dans la composition d'une autre un certain nombre de fois.

ex : La pièce "voiture" est composée de 4 pièces "roue". La pièce "roue" est elle-même composée d'une pièce "pneu" et d'une pièce "jante".

Une pièce entrant dans la composition d'une autre est appelée composant. Une pièce composée d'autres pièces est appelée composé. Une roue est à la fois un composant (de voiture) et un composé (de pneu et jante)

Traduction en modèle relationnel

PIECE(référence, libellé)COMPOSITION(#référence_composé, #référence_composant, nombre)