

1. Concepts de Base

Définition 1 : Un domaine est un ensemble de valeurs atomiques.

Définition 2 : Le produit cartésien $D_1 \times D_2 \times \dots \times D_n$ est l'ensemble des tuples (n-uplets) $\langle V_1, V_2, \dots, V_n \rangle$ tel que quel que soit i V_i appartient à D_i .

Exemple :

$D_1 = \{\text{Bleu}, \text{Blanc}, \text{Rouge}\}$, $D_2 = \{\text{Vrai}, \text{Faux}\}$.

$D_1 \times D_2 = (\text{Bleu}, \text{Vrai}); (\text{Bleu}, \text{Faux}); (\text{Blanc}, \text{Vrai}); (\text{Blanc}, \text{Faux});$
 $(\text{Rouge}, \text{Vrai}); (\text{Rouge}, \text{Faux})$

Définition : Une relation est un sous-ensemble nommé du produit cartésien d'une liste de domaines. elle est notée $R(A_1:D_1, \dots, A_n:D_n)$ où D_1, \dots, D_n sont des domaines.

Note : On peut également noter la relation sans mentionner les domaines : $R(A_1, A_2, \dots, A_n)$

Exemple

$D_1 = \text{COULEUR}$

$D_2 = \text{BOOLEEN}$

Couleur_Véhicule(Couleur : D_1 , Existe: D_2)

Couleur_Véhicule	Couleur	Existe
	Bleu	FAUX
	Blanc	VRAI
	Rouge	VRAI

Note : L'extension d'une relation est variable au cours de la vie d'une base de données.

Vision tabulaire du relationnel

- Une relation est une table à deux dimensions,
- Une ligne est un tuple,
- Un nom est associé à chaque colonne afin de la repérer indépendamment de son numéro d'ordre.

Vision Assertionnelle

A toute relation de schéma $R(A_1:D_1, \dots, A_n:D_n)$ est associé un prédicat R tel que l'assertion $R\ t$ est vraie si le n-uplet t appartient à l'extension de R et fausse sinon.

On suppose la relation de schéma: Personne (Nom : Chaîne, Age : Entier, Marié : Booléen)
 Et d'extension : {{Nom="Hakim" , Age=23 , Marié = Faux};
 {Nom="Lila" , Age=36 , Marié = Vrai } }

Assertionnelle

Personne{Nom="Hakim", Age=23, Marié=Faux} **Vrai**
 Personne{Nom="Lila", Age=35, Marié=Vrai} **Faux**
 Personne{Nom="Lila", Age=23, Marié=Faux} **Faux**

Tabulaire

Nom	Age	Marié
Hakim	23	Faux
Lila	36	Vrai

Attribut

Un attribut est un nom donné à une colonne d'une relation, il prend ses valeurs dans un domaine.

Exemple : Nom, Age et Marié sont des attributs de Personne

Clé d'une relation

Définition : C'est un groupe d'attributs minimum qui détermine un tuple unique dans une relation.

Schéma d'une relation

Définition : Nom de la relation, liste des attributs avec domaines, et liste des clés d'une relation.

Exemples: Voiture(Modèle: texte, Année:Ent, Couleur:texte, Type: texte)

Intention et extension d'une relation : Un schéma de relation définit l'intention de la relation alors qu'une instance de table représente une extension de la relation.

Clé étrangère

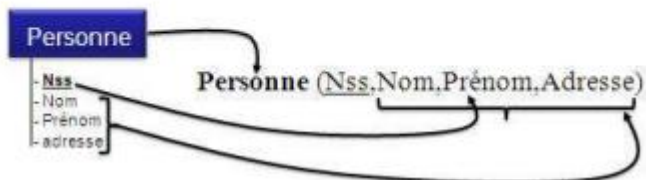
Définition : Groupe d'attributs devant apparaître comme clé dans une autre relation.

Les clés étrangères définissent les contraintes d'intégrité référentielles suivantes:

2. Passage d'entité – association au modèle relationnel

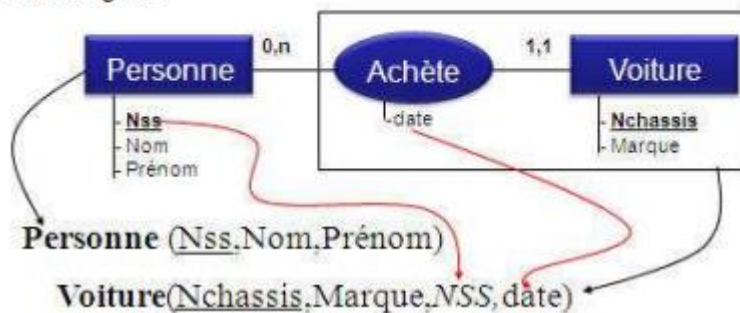
Règle 1 : Entité non faible

Une entité E non faible est représentée par une relation T dont les attributs simples sont les attributs de l'entité E et la clé de T est l'identifiant de E.



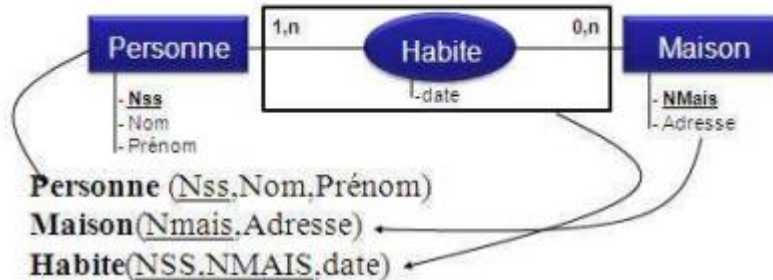
Règle 2 : Relation 1 : n

Dans le cas d'une relation 1:n (Père/fils), l'association n'est pas représentée par une relation, cependant ses attributs migrent vers la relation représentant le fils et la clé du père migre vers le fils comme clé étrangère.



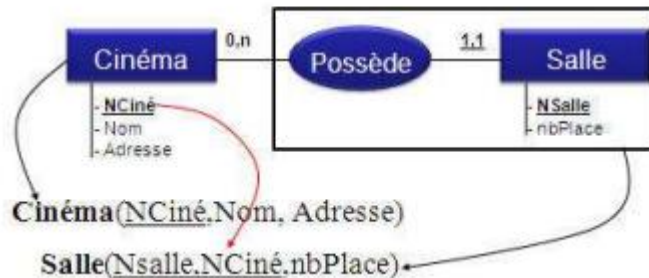
Règle 3 : Relation n : m

Dans le cas d'une relation n:m, l'association A est représentée par une relation T dont les attributs sont les attributs de A et la clé est la concaténation des clés des entités participant à l'association A.



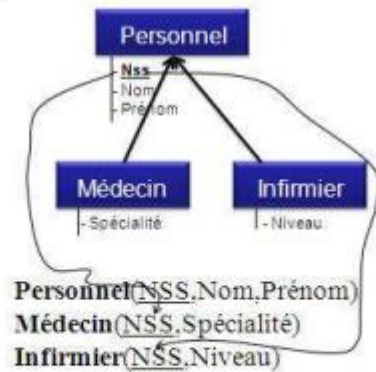
Règle 4 : Entité faible

Le passage d'une entité faible à un schéma relationnel est identique à celui d'une association 1-n classique. La seule différence est que la clé du père migre vers le fils est entre dans la constitution de la clé primaire de ce dernier.

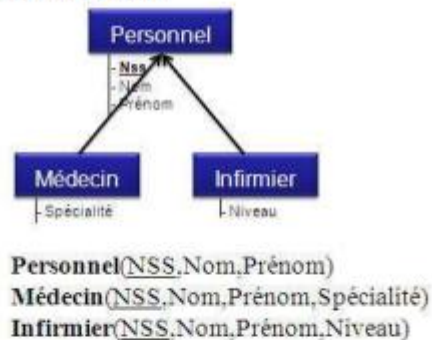


Règle 5 : Généralisation/spécialisation

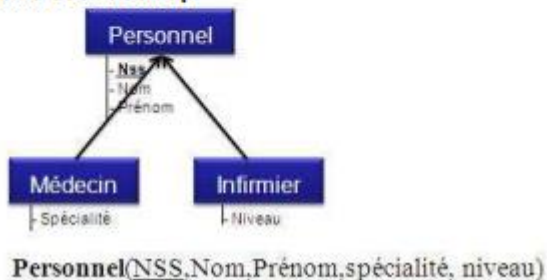
Premier Cas :



Deuxième Cas : Push down



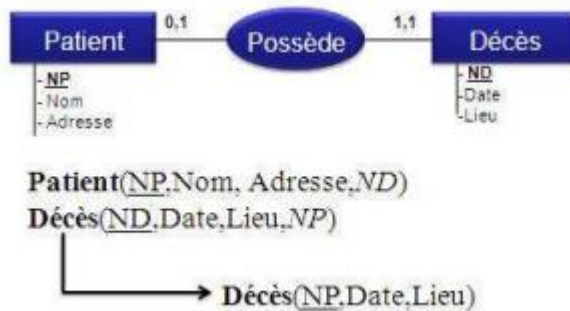
Troisième Cas : Push up



Règle 5 : Association 1 :1

Dans le cas d'une association 1:1, la migration de la clé peut se faire dans un seul sens comme on peut le faire dans les deux sens. il est à noter que quand un cas pareil se présente il est presque

impossible que les cardinalités minimales des deux côtés soit toutes les deux égales à 1, l'une d'elles est toujours un 0. Ce qui fait que le sens le plus recommandé pour la migration de la clé est celui allant du côté de la cardinalité minimale 0 vers celui de la cardinalité minimale 1.



3. Dépendance fonctionnelles

- ✓ Il existe une dépendance fonctionnelle de A vers B
- ✓ A détermine B
- ✓ B dépend fonctionnellement de A
- ✓ A source de dépendance fonctionnelle et B sa destination

$A \rightarrow B$

DF élémentaires

$A \rightarrow B$ élémentaire \Rightarrow il n'existe pas un C inclus dans A qui assure une dépendance fonctionnelle $C \rightarrow B$

1. Ref_Article \rightarrow Nom_Article

2. Num_Facture, Ref_Article \rightarrow Qte_Article

3. Num_Facture, Ref_Article \rightarrow Nom_Article

Les dépendances fonctionnelles 1 et 2 sont élémentaires alors que la 3 ne l'est pas. parce qu'il existe une partie de la source de la df qui assure elle-même la dépendance fonctionnelle : c'est Ref_Article au niveau de la df 1.

DF directe

Il n'existe pas un attribut C qui engendre une dépendance fonctionnelle transitive $A \rightarrow C \rightarrow B$

1. Num_Facture \rightarrow Num_Représentant VRAI

2. Num_Représentant \rightarrow Nom_Représentant VRAI

3. Num_Facture \rightarrow Nom_Représentant FAUX

Les dépendances fonctionnelles 1 et 2 sont directes alors que la troisième ne l'est pas parce qu'elle peut être déduite par transitivité à partir des deux premières.

DF triviales

Une dépendance fonctionnelle est dite triviale s'il est impossible qu'elle ne soit pas satisfaite. Une dépendance fonctionnelle est triviale si et seulement si le membre droit (destination) est un sous-ensemble du membre gauche (source).

$A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$

Triviale: Si B_1, B_2, \dots, B_m est un sous-ensemble de A_1, A_2, \dots, A_n

Non triviale : Si au moins un B_i n'appartient pas à A_1, A_2, \dots, A_n

Complètement non triviale : Si aucun des B_i n'appartient à A_1, A_2, \dots, A_n