

Bases de Données & Technologies Web

INFO642

Normalisation

Le modèle relationnel (suite)

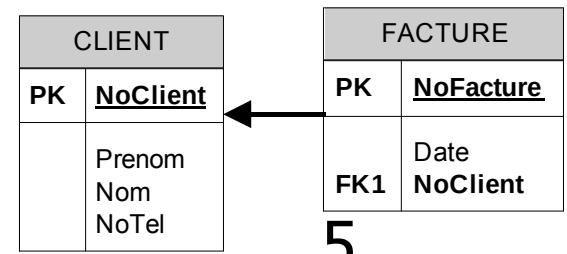
- 2 propriétés des tuples à respecter
 - L'unicité des tuples : il ne peut y avoir de tuples identiques
 - L'ordre des tuples : l'ordre des tuples n'a pas d'importance, c'est la même occurrence
- 3 propriétés des attributs à respecter
 - Indivisibilité : Les données ne sont pas décomposables
 - Domaine unique : les attributs ne peuvent prendre n'importe quelle valeur (intervalle, type de données)
 - Ordre : l'ordre des attributs n'a pas d'importance

Dépendance fonctionnelle (DF)

- $X \rightarrow Y$
 - Y est en dépendance fonctionnelle de X...
 - X détermine fonctionnellement Y...
 - ... si et seulement si à chaque valeur de X correspond exactement une seule valeur de Y
 - Autrement dit, lorsque 2 tuples s'accordent sur leur valeur de X, ils s'accordent également sur leur valeur de Y

Dépendance fonctionnelle (DF)

- Les dépendances fonctionnelles peuvent être internes
 - DF entre attributs d'une même entité
 - Ex. : PERSONNE (NSS, Nom, Prénom)
 - NSS → Nom
 - Avec le NSS, je trouve un et un seul nom
- Elles peuvent aussi être externes
 - DF entre entités
 - Facture → Client



La normalisation

- Théorie élaborée par E.F. Codd en 1970
- But : éviter les anomalies dans les bases de données relationnelles
 - Supprimer certains types de redondances
 - Éviter certaines anomalies de mise à jour
 - Élaborer une conception représentative du monde réel
 - Simplifier la satisfaction de certaines contraintes d'intégrité
- Plus le niveau des formes normales est élevé pour une table, plus cette table sera exempte d'anomalies
- Un bon MCD implique souvent un modèle relationnel ayant déjà atteint un bon niveau de normalisation

Première forme normale (1FN)

- Une table est en 1FN si tous ses attributs sont simples (non décomposables), domaine unique par attribut
- Ne sont pas en 1FN :
 - PERSONNE (NSS, Nom, Prénom, PrénomEnfants)
 - PERSONNE (NSS, Nom, Prénom, Adresse)
- Si la transposition d'un MCD en modèle relationnel n'est pas déjà en 1FN, c'est qu'il a été mal modélisé !

Première forme normale (1FN)

- PERSONNE (Nom, Prénom, Adresse, Ville, Département)
 - Exemple,

Nom	Adresse	Ville
Durand	jean.d@gmail.com	Chambéry 73
Paul Dupont	45 avenue Pasteur	74
Olivier Michel	123 bis rue Lavoisier	Paris



Nom	Prénom	Adresse	Ville	Département
Durand	Jean	21 rue des quais	Chambéry	73
Dupont	Paul	45 avenue Pasteur	Annecy	74
Michel	Olivier	123 bis rue Lavoisier	Paris	75

Première forme normale (1FN)

- PERSONNE (Nom, Prénom, prénomEnfants) non 1FN

Solution partielle (1 enfant 1 parent représentant) :

- PERSONNE (Nom, Prénom)
- PRENOMSENFANT(#(NomParent, PrénomParen),
rangPrénomEnfant, prénomEnfant)

Solution complète (1 enfant 2 parents : 3 personnes) :

- PERSONNE(IdPersonne, NomPersonne)
- PRENOMS(#IdPersonne, rangPrénom, prénom)
- est_enfant_de(#IdPersonne, #IdPersonne1, #IdPersonne2)

Deuxième forme normale (2FN)

- Une table est en 2FN si et seulement si elle est en 1FN et si chaque attribut non clé ne dépend pas d'une partie de la clé primaire
- Autrement dit, une table est en 2FN si
 - Elle est en 1FN
 - Une des 3 conditions suivantes est vérifiée
 - La clé primaire n'est formée que d'un seul attribut
 - La clé primaire contient tous les attributs de la table
 - Si la clé primaire a plus d'un attribut, une dépendance fonctionnelle ne doit jamais exister entre une partie de la clé et un autre attribut de la table
 - Tout attribut qui ne fait pas partie de la clé dépend de toute la clé (dépendance fonctionnelle totale)

Deuxième forme normale (2FN)

- Pour passer de la 1FN à la 2FN, il faut diviser chaque table ne satisfaisant pas les critères en deux tables distinctes
- Pour diviser une table en deux, il faut
 - Créer une nouvelle table ayant pour clé la partie de la clé primaire dont dépend le ou les attributs, ainsi que ces attributs eux-mêmes
 - Éliminer ces attributs (ceux qui ne font pas partie de la clé) de la table originale

Deuxième forme normale (2FN)

- SALARIE (NumSalarié, Nom)
 - ~~PROJET(NumProjet)~~ - - optimisation
 - travaille(#NumSalarié, #NumProjet, Heures)
- Exemple,

NumSalarié	Nom	NumProjet	Heures
39002	Durand	1	20
39002	Durand	2	7
78291	Dupont	2	9
42778	Michel	3	16
42778	Michel	1	5

The diagram illustrates a star-shaped relationship between three tables. A central triangle has arrows pointing from each vertex to one of the three tables: SALARIE (left), PROJET (right), and travaille (bottom). The SALARIE table contains columns NumSalarié and Nom. The PROJET table contains columns NumProjet and Heures. The travaille table contains columns NumSalarié, NumProjet, and Heures.

NumSalarié	Nom	NumSalarié	NumProjet	Heures
39002	Durand	39002	1	20
78291	Dupont	39002	2	7
42778	Michel	78291	2	9
		42778	3	16
		42778	1	5

Relation (jointure)

Troisième forme normale (3FN)

- Une table est en 3FN si et seulement si elle est en 2FN et si chaque attribut non clé est en dépendance non transitive avec la clé primaire
- Autrement dit, une table est en 3FN si
 - Elle est en 2FN
 - Aucun attribut ne faisant pas partie de la clé primaire ne dépend d'un autre attribut ne faisant pas partie lui non plus de la clé primaire
 - Les dépendances fonctionnelles entre deux attributs ordinaires (ne faisant pas partie de la clé) sont interdites

Troisième forme normale (3FN)

- Pour passer de 2FN à 3FN, il faut
 - Diviser chaque table ne satisfaisant pas ce critère en deux tables. La nouvelle table aura comme clé l'attribut dont provient la dépendance et comme attributs, ceux qui en dépendent.
 - Éliminer les attributs dépendants de la table originale. La clé de la nouvelle table demeure dans l'ancienne en tant que clé étrangère

Troisième forme normale (3FN)

- SALARIE (NumSalarié, Nom, DateNaiss., #NumService)
- SERVICE (NumService, NomService, #NumChef)

- Exemple,

Nom	NumSalarié	Date Naiss.	Service	NomService	NumChef
Durand	39002	17/08/1981	4	Comptabilité	21987
Dupont	78291	30/04/1973	7	Vente	56720

NumSalarié	Nom	Date Naiss.	Service
39002	Durand	17/08/1981	4
78291	Dupont	30/04/1973	7

Relation (jointure)



Service	Nom	NumChef
4	Comptabilité	219873
7	Vente	56720

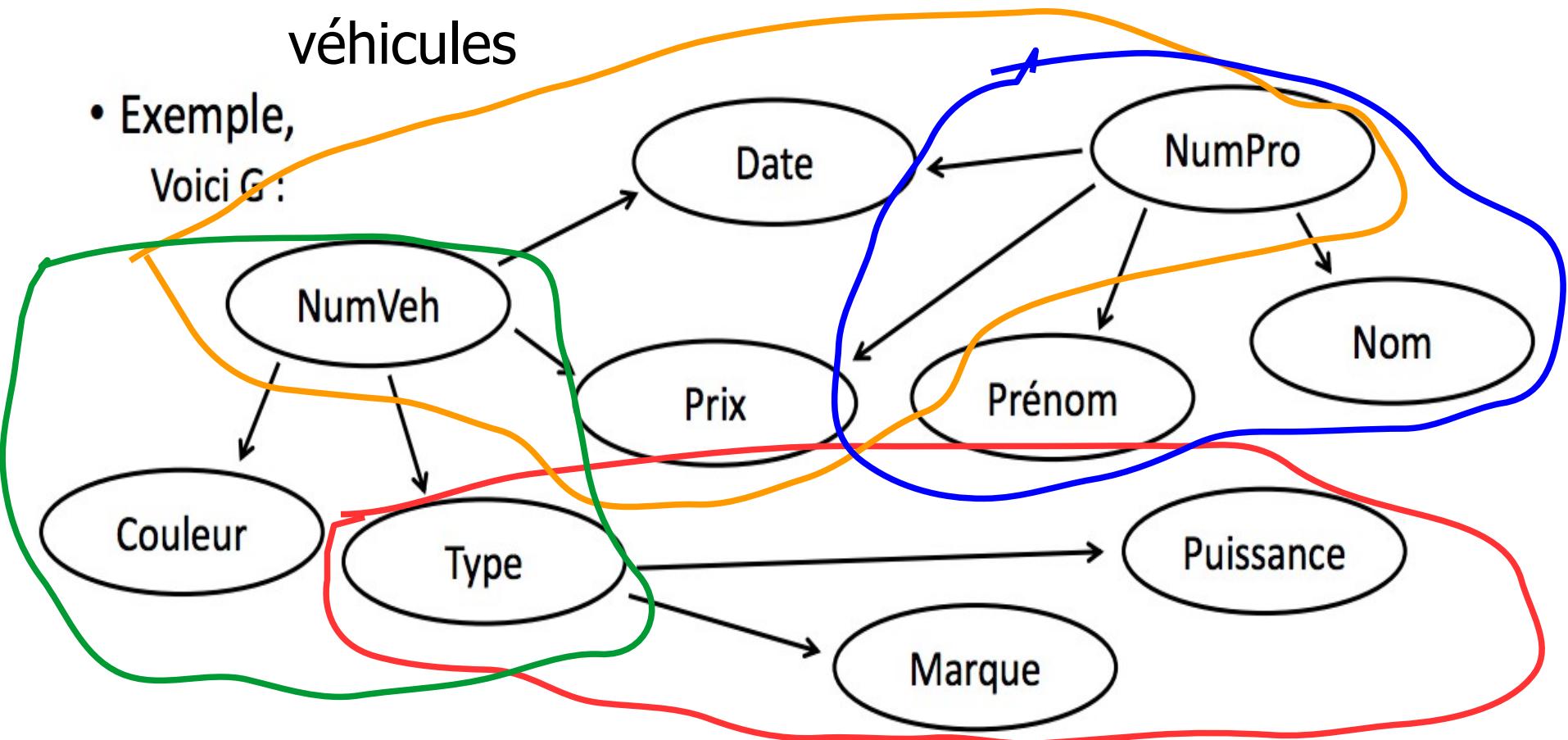
RQ. : pk et fk peuvent avoir des noms différents

Dépendances dans les 2 sens : ajouter les contraintes référentielles après la création des tables

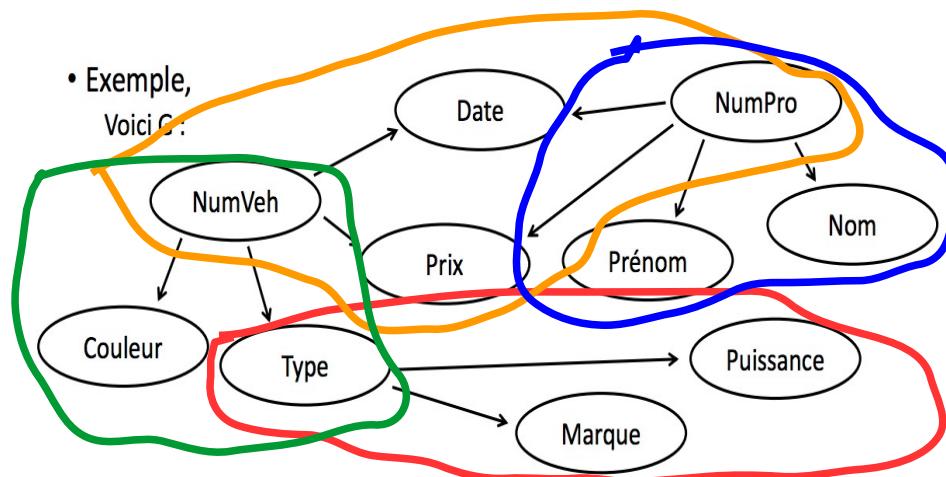
Graphe de dépendances et Normalisation : décomposition

Une relation sur les véhicules

- Exemple,
Voici G :



Graphe de dépendances et Normalisation : décomposition



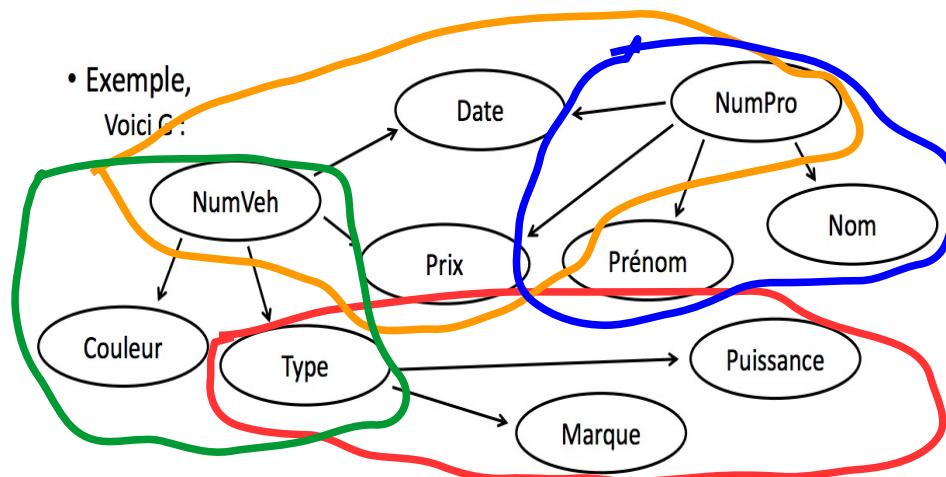
Véhicule (NumVeh, Couleur, #Type)

Modèle (Type, Puissance, Marque)

Personne (NumPro, Nom, Prénom)

Possession (#NumVeh, #NumPro, Date, Prix)

Graphe de dépendances et Normalisation : décomposition



Véhicule (NumVeh, Couleur, #Type)

Modèle (Type, Puissance, Marque)

Personne (NumPro, Nom, Prénom)

Possession (#NumVeh, #NumPro, Date, Prix)

Forme normale de Boyce-Codd (FNBC)

- Définition plus rigide de la 3FN
- Une table est en FNBC si
 - Elle est en 3FN
 - Aucun attribut faisant partie de la clé primaire ne dépend d'un attribut ne faisant pas partie de la clé primaire (2FN en sens inverse)
- Autrement dit, une table est en FNBC si une des conditions suivantes est remplie
 - Sa clé n'est constituée que d'un seul attribut
 - Tous les attributs sont dans la clé primaire
 - Il n'y a pas de dépendance fonctionnelle entre un attribut ordinaire et un attribut faisant partie de la clé primaire

Forme normale de Boyce-Codd (FNBC)

- Une base de donnée peut généralement être considérée comme implantable lorsqu'elle est en FNBC
- Les cas de tables modélisées et transformées en 3FN qui ne sont pas déjà en FNBC sont très rares