

Modélisation Logique des Données

version à compléter

IUT de Bayonne et du Pays Basque

BUT Informatique – Semestre 1

Ressource R1.05 « *Introduction aux Bases de Données et SQL* » – Partie Conception

Année Universitaire 2021-2022

Christophe MARQUESUZAÀ

Christophe.Marquesuzaa@iutbayonne.univ-pau.fr

Bureau 202



IUT de Bayonne et du Pays Basque, Année Universitaire 2021-2022, BUT Informatique, R1.05 (Conception), Christophe Marquesuzaa

2


Bibliographie

- Supports de cours de collègues
- Pantxika DAGORRET
 - IUT de Bayonne et du Pays Basque, Département Informatique
- Bernard MORAND
 - IUT de Caen, Département Informatique

Plan

- Niveaux d'abstraction de la modélisation
- Niveau logique – Concepts et Exemple
- Opérations de manipulation de données
- Règles de transformation
- Conclusion

Niveaux d'abstraction de la modélisation

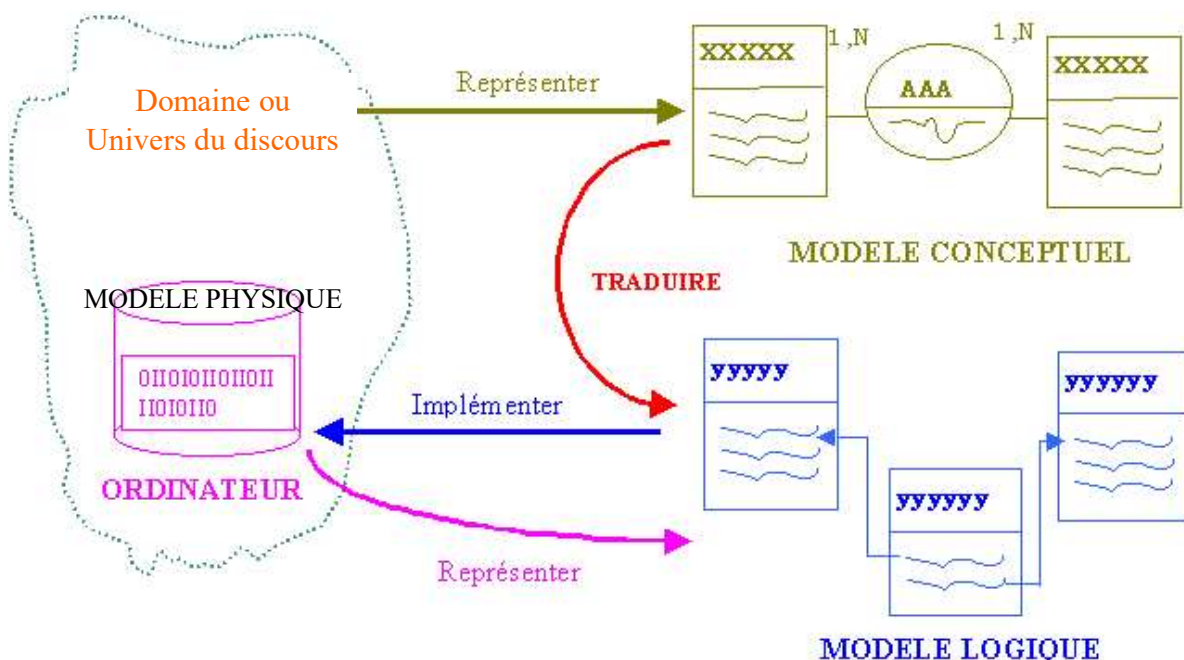
- Un  de données (MCD MERISE ou Diagramme de Classes UML) est indépendant de l'état de l'art technologique.
- Il existe différentes sortes de SGBD qui ont chacun leur propre (dans l'ordre historique) :
 - Systèmes de Gestion de Fichiers (SGF) qui ne sont pas vraiment des SGBD.
 - SGBD hiérarchiques organisés selon une arborescence.
 - SGBD réseau encore appelés CODASYL (du nom de la norme à laquelle ils se réfèrent).
 - : Oracle, Access, MySQL, PostgreSQL, DB2, etc.
 - SGBDOO (Orientés Objet) : O2, Jasmine, Versant, DB40, etc. *Certains SGBDR disposent aussi d'une extension objet.*
 - SGBD XML : Tamino, eXist, etc. *Certains SGBDR disposent aussi d'une extension XML.*
- On ne sait pas implémenter aisément un modèle « conceptuel » dans une machine :
 - Une association de dimension supérieure à 2 doit être transformée.
 - Une association porteuse de données n'est pas toujours implémentable telle quelle.

Niveaux d'abstraction de la modélisation

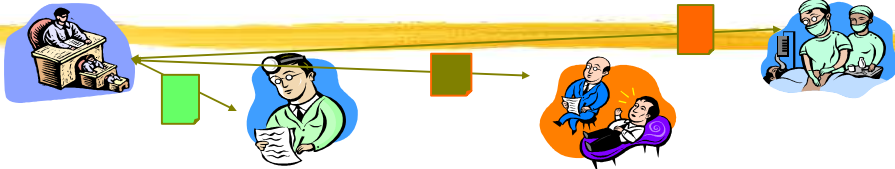
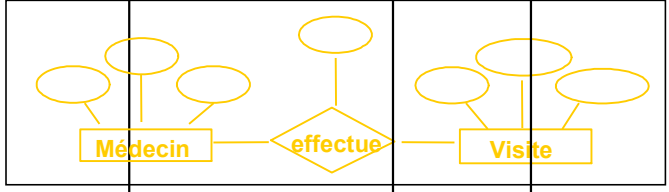
- Le est une du
système tel qu'il sera implanté.
- Vu qu'il s'agit d'une représentation, on utilisera également un modèle !
- Toutefois :
 - Il ne faut pas confondre le modèle « conceptuel » (entité - association par exemple) avec le modèle « logique » (relationnel par exemple).
 - Il ne faut pas confondre le modèle « logique » (relationnel par exemple) avec son implémentation « physique » en machine (avec Access, MySQL, PostgreSQL ou Oracle par exemple).

Niveaux d'abstraction de la modélisation

- On passe du modèle conceptuel au modèle logique par une opération de



Niveaux d'abstraction de la modélisation

Réel (<i>monde physique</i>)				
Modèle conceptuel	<input type="checkbox"/> Indépendant du SGBD			
Modèle logique	<input type="checkbox"/> Dépendant du modèle conceptuel <input type="checkbox"/> Indépendant du SGBD	Codasyl	Relationnel (<i>textuel ou graphique</i>)	Objet XML
Modèle physique	<input type="checkbox"/> Dépendant du SGBD	<input type="checkbox"/> Organisation physique des données <input type="checkbox"/> Structures de stockage des données <input type="checkbox"/> Structures accélératrices (index)		

7

Niveau logique – Concepts

- Une **relation** est une partie finie du produit cartésien de n ensembles (ou **domaines**).
- Un **attribut** prend ses valeurs dans un **ensemble** (soit l'ensemble de ses valeurs possibles). Ainsi, une **relation** (ou **table**) est le résultat d'une opération sur les ensembles constitués par les valeurs des attributs.
- Attribut de relation** : un attribut particulier (ou un ensemble d'attributs) tel qu'il ne peut exister qu'une seule valeur de cet attribut pour tous les n -uplets de la table. Ce pourrait être le cas (dans le transparent suivant) de l'attribut « code » de la table « CLIENT » (qui ressemble à notre définition d'identifiant au niveau conceptuel).
- Un **enregistrement** est un enregistrement (une ligne) dans la base de données.
- Cardinalité** de relation : nombre de ses **occurrences** (ou occurrences, enregistrements).
- Largeur** (ou encore **nombre d'attributs**) de relation : nombre des attributs (ou colonnes). Rien à voir donc avec la **cardinalité** (nombre d'entités reliées) d'une association.

Niveau logique – Exemple

- Exemple de TABLE (RELATION) « CLIENT » :

« relation en extension »

code	nom	adresse	ville
AXV508	Dupont	15, rue du Canada	Caen
BWY405	Durand	6, rue Méchin	Cherbourg
KAC567	Duval	25, rue Blanchi	Alençon

4 attributs

3 n-uplets ou tuples

Valeurs d'attributs

- Schéma (textuel) d'une relation (« relation en intention ») :

CLIENT (code, nom, prenom, adresse)

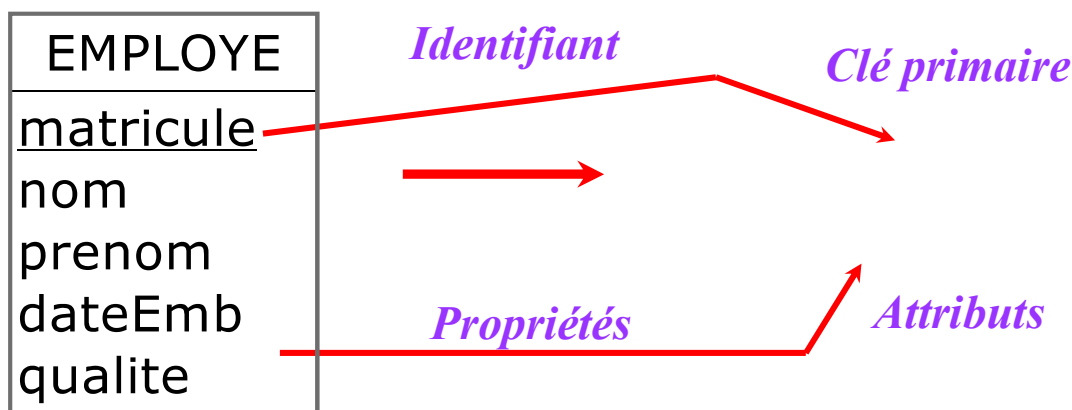
Nom de relation Clé primaire de relation Attributs

Opérations de manipulation de données

- Langage SQL (*Structured Query Language*)
- Cf. enseignements de Manipulation de Bases de Données
 - Sélection
 - Projection
 - Jointure
 - Union
 - Intersection
 - Différence
 - Etc.

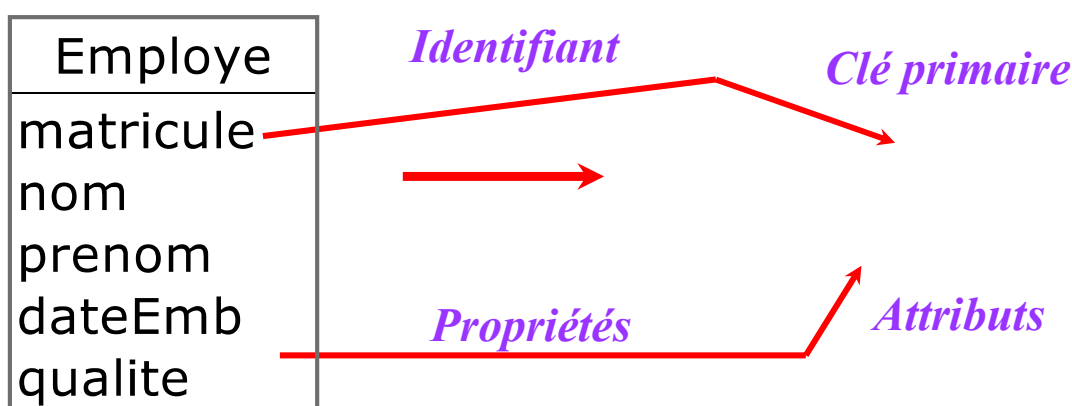
Règles de passage du Modèle Conceptuel (formalisme MERISE) au Schéma Relationnel

- Règle n°1 : Toute (classe d')entité devient une relation dans laquelle :
 - les *attributs* traduisent les *propriétés* de l'entité
 - la *clé primaire* traduit *l'identifiant* de l'entité



Règles de passage du Modèle Conceptuel (formalisme UML) au Schéma Relationnel

- Règle n°1 : Toute (classe d')entité devient une relation dans laquelle :
 - les *attributs* traduisent les *propriétés* de l'entité
 - la *clé primaire* traduit *l'identifiant* de l'entité



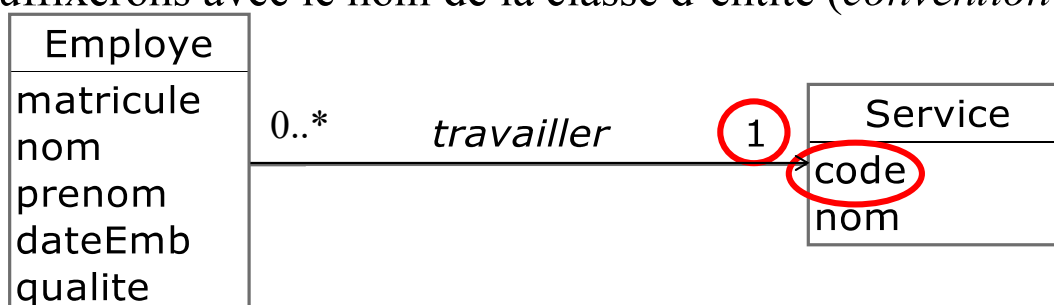
Règles de passage du Modèle Conceptuel (formalisme MERISE) au Schéma Relationnel

- Règle n°2 : Une (classe d')association de dimension 2 avec cardinalité 1,1 (ou 0,1) se réécrit en :
 - portant dans la **relation fille** la clé primaire de la **relation mère**.
 - L'**attribut** ainsi **ajouté** s'appelle **clé étrangère**. Par convention, nous la symboliserons au moyen de # (*Attention, ce caractère ne fait toutefois pas partie du nom du champ dans la BD !*) et nous suffixerons avec le nom de la classe d'entité (*convention d'écriture*).



Règles de passage du Modèle Conceptuel (formalisme UML) au Schéma Relationnel

- Règle n°2 : Une (classe d')association de dimension 2 avec multiplicité 1 (ou 0..1) se réécrit en :
 - portant dans la **relation fille** la clé primaire de la **relation mère**.
 - L'**attribut** ainsi **ajouté** s'appelle **clé étrangère**. Par convention, nous la symboliserons au moyen de # (*Attention, ce caractère ne fait toutefois pas partie du nom du champ dans la BD !*) et nous suffixerons avec le nom de la classe d'entité (*convention d'écriture*).



Règles de passage du Modèle Conceptuel au Schéma Relationnel

■ Règle n°2 : Recette

- On hésite quelquefois au début pour savoir dans laquelle des 2 relations, il faut porter la clé étrangère. Il faut contrôler si cet attribut est bien monovalué (*i.e.* qu'il respecte bien la 1ère Forme Normale).
- Supposons que, par erreur, nous portions « matricule » dans SERVICE comme clé étrangère. Si tel est le cas, alors cet attribut aurait comme valeurs tous les matricules des employés de ce service !
- Par contre, dans la solution correcte de l'exemple précédent, on voit clairement que **#codeService** ne peut posséder qu'une valeur, celle du service dans lequel travaille cet employé (il n'y en a qu'un).

Règles de passage du Modèle Conceptuel au Schéma Relationnel

■ Cas particulier de la cardinalité 0,1 / multiplicité 0..1

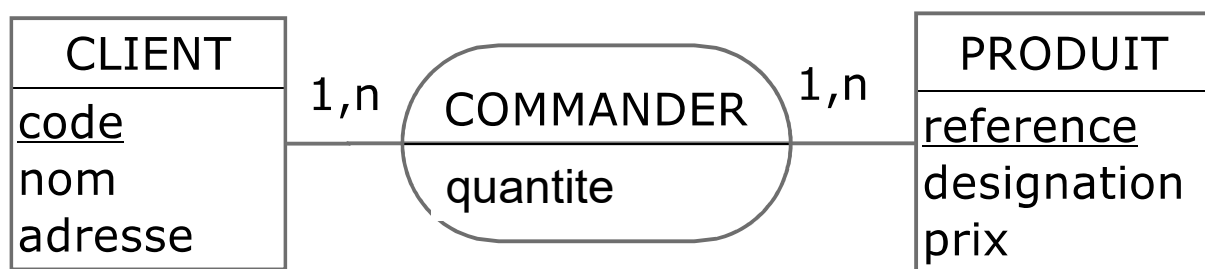
- La même règle s'applique mais la clé étrangère pourra avoir une « valeur nulle ». Ainsi, si un employé peut ne pas travailler dans un service, alors il se peut que **#codeService** n'ait pas de valeur. Il en résulte que l'on ne peut plus, dans ce cas, utiliser aisément le mécanisme de vérification du respect de l'intégrité référentielle.

■ Remarques

- Nous avons mentionné qu'une classe d'association avec une cardinalité 1,1 / multiplicité 1 ne devrait pas être porteuse de propriétés. On vérifie maintenant que si cela a toutefois été le cas au niveau conceptuel, alors cette propriété devient attribut de la relation enfant.
- Attention aussi aux classes d'association ayant un minimum de 1 pour toutes les cardinalités/multiplicités (dimension n:m). Cela peut poser des contraintes lors du (dé)peuplement et nécessiter des outils de gestion évolués (triggers par exemple).

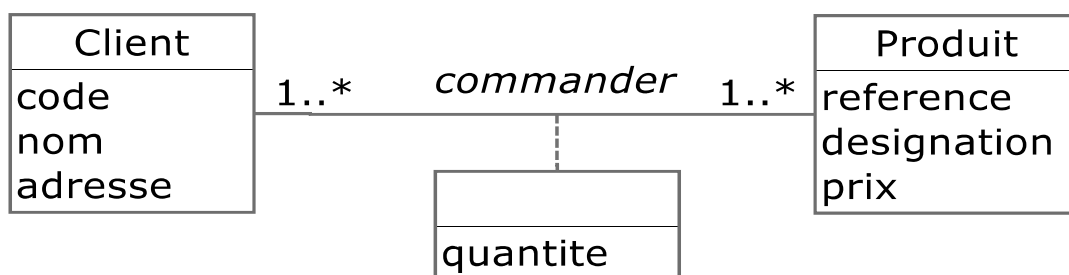
Règles de passage du Modèle Conceptuel (formalisme MERISE) au Schéma Relationnel

- Règle n°3 : Une (classe d')association de dimension 2 avec des cardinalités 1,n et-ou 0,n se réécrit en :
 - créant une relation particulière qui contient comme attributs les identifiants des 2 (classes d')entités associées
 - ces attributs constituent à eux 2 la clé primaire de la relation
 - ils sont individuellement clés étrangères
 - ajoutant la ou les éventuelles propriétés de l'association comme attributs de cette relation.



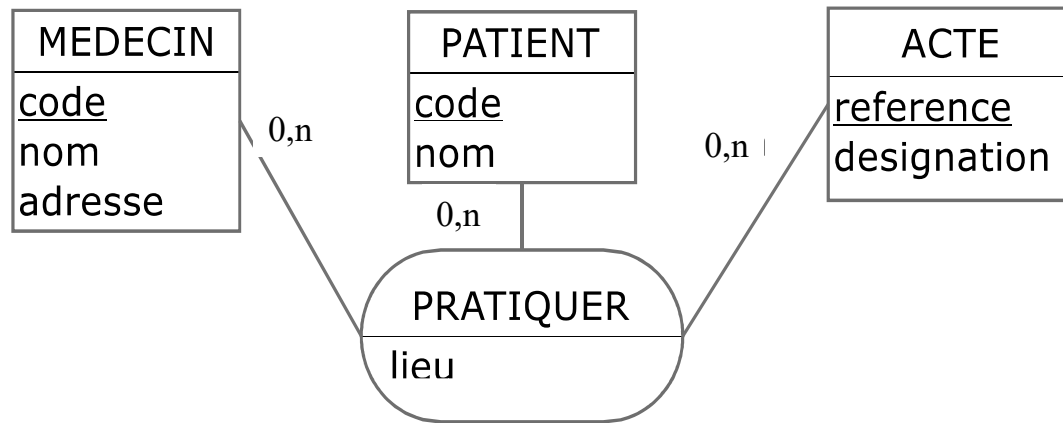
Règles de passage du Modèle Conceptuel (formalisme UML) au Schéma Relationnel

- Règle n°3 : Une (classe d')association de dimension 2 avec des multiplicités 1..* et-ou 0..* se réécrit en :
 - créant une relation particulière qui contient comme attributs les identifiants des 2 (classes d')entités associées
 - ces attributs constituent à eux 2 la clé primaire de la relation
 - ils sont individuellement clés étrangères
 - ajoutant la ou les éventuelles propriétés de l'association comme attributs de cette relation.



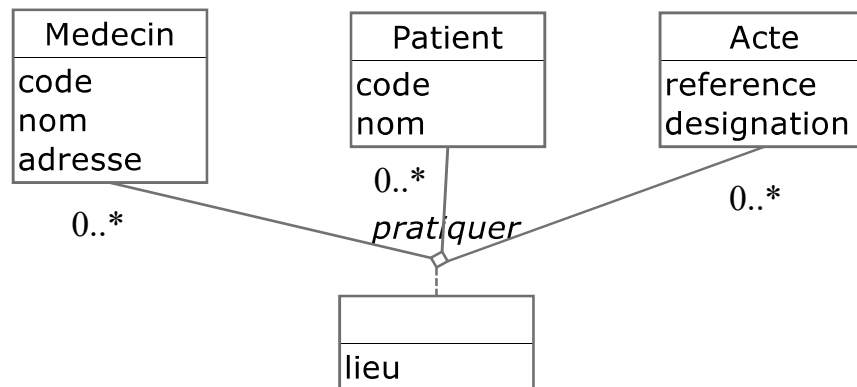
Règles de passage du Modèle Conceptuel (formalisme MERISE) au Schéma Relationnel

- Règle n°4 : Une (classe d')association de dimension supérieure à 2 se réécrit selon la règle n°3.



Règles de passage du Modèle Conceptuel (formalisme UML) au Schéma Relationnel

- Règle n°4 : Une (classe d')association de dimension supérieure à 2 se réécrit selon la règle n°3.



Conclusion

- Le schéma relationnel peut se présenter sous **forme textuelle**, comme dans ce document, en respectant des normes/conventions d'écriture (exemple : nom de table en majuscule, clé primaire soulignée, clé étrangère précédée de #) mais aussi sous **forme graphique** (comme dans MS Access par exemple).
- **Ne pas oublier de mettre à jour le dictionnaire des données** en :
 - ajoutant les attributs qui sont les clés étrangères.
 - ajoutant/renseignant une colonne pour le type physique (en précisant le SGBD choisi/imposé)
- En conclusion, les règles de passage d'un modèle de niveau conceptuel en niveau logique puis physique sont totalement formalisables. Par conséquent, on peut les automatiser au moyen d'un
 - Un AGL vous aide ainsi à dessiner votre modèle E-A et génère ensuite automatiquement le schéma de la BD ; cela pouvant fonctionner dans les 2 sens (rétro-ingénierie).
 - Exemples d'AGL : WinDev de PCSoft, AMCDesignor de Sybase, Rational Rose d'IBM, Modelio, BOUML, ArgoUML, **MySQL Workbench** (ex DBDesigner), **Looping** (<https://www.looping-mcd.fr/>), etc.

Conclusion

- Le **Dictionnaire des Données (mis à jour)** est aussi utile pour estimer le **calcul des volumes de données** nécessaires à la mise en place de la base de données relationnelle grâce à une estimation du nombre d'enregistrements de chacune des tables. Ceci est particulièrement pertinent en informatique embarquée ou pour les gros volumes.
- Exemple pour la table EMPLOYE :

Libellé	Signification	Domaine			Contraintes	Taille	Exemple
		niveau conceptuel	niveau logique	niveau physique (Oracle)			
code	Code identifiant chaque employé	numérique	entier	NUMBER	Clé primaire	5	12345
nom	Patronyme de chaque employé	alphabétique	chaîne de caractères	VARCHAR2		25	KükürütX
codeService	Code du service pour lequel travaille chaque employé	alphanumér.	chaîne de caractères	VARCHAR2	Clé étrangère faisant référence au code unique du Service. Obligatoirement renseigné.	6	info02
...							

- Si on suppose que un entier est codé sur un octet, un caractère sur un octet, chaque enregistrement de employé sera codé sur 37 octets. Si on suppose que notre SI traitera 1000 employés, notre SI nécessitera au minimum 37 Ko et ainsi de suite...

Conclusion

- Quelques remarques de vocabulaire.
- Le « *Domaine* » est encore appelé « *Univers du discours* ».
- On parle aujourd'hui « *d'objets métiers* » ou encore « *d'ingénierie des besoins* ».
- Vous devez ici faire « naturellement » le lien avec vos cours de (manipulation de) Bases de Données. Au niveau physique, le vocabulaire dépend grandement du SGBDR choisi.

Niveau conceptuel	Niveau logique	Niveau physique
Classe d'entité		
Classe d'association		
Propriété		
Identifiant		
Occurrence		

Notre processus de développement simplifié

