

# Plan UE

## Introduction aux Bases de Données



Généralités sur les  
BDD

- BDD, SGBD, MCD

Le modèle  
logique  
relationnel

- Concepts, Passage, Intégrité référentielle

Interrogation des  
BDD & langage SQL

- Algèbre relationnelle
- SQL

# CHAPITRE 2



## LE MODÈLE LOGIQUE DE DONNÉES MODÈLE RELATIONNEL

# Plan



- **MODÈLE LOGIQUE DE DONNÉES**

- **MODÈLE RELATIONNEL**

Généralités

Concepts fondamentaux

- **PASSAGE MCD VERS MLD**

Règles de passage

Exemples

- **CONTRAINTES D'INTÉGRITÉ**

# I/Modèle Logique de Données

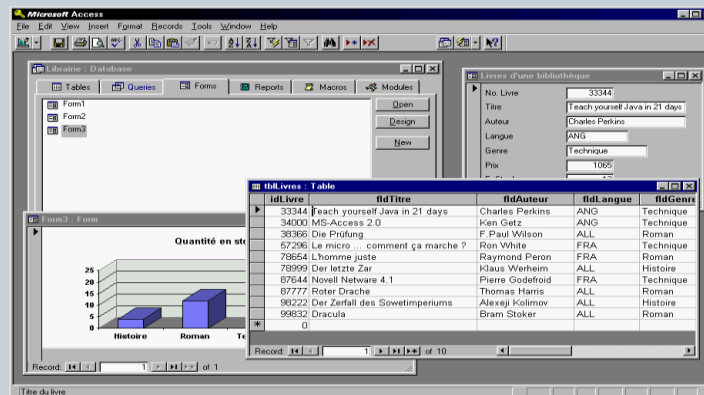
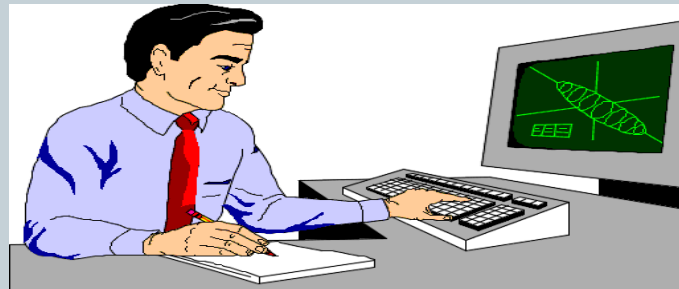
Analyse (entreprise, documents.....)

MCD

MLD

MPD

BDD



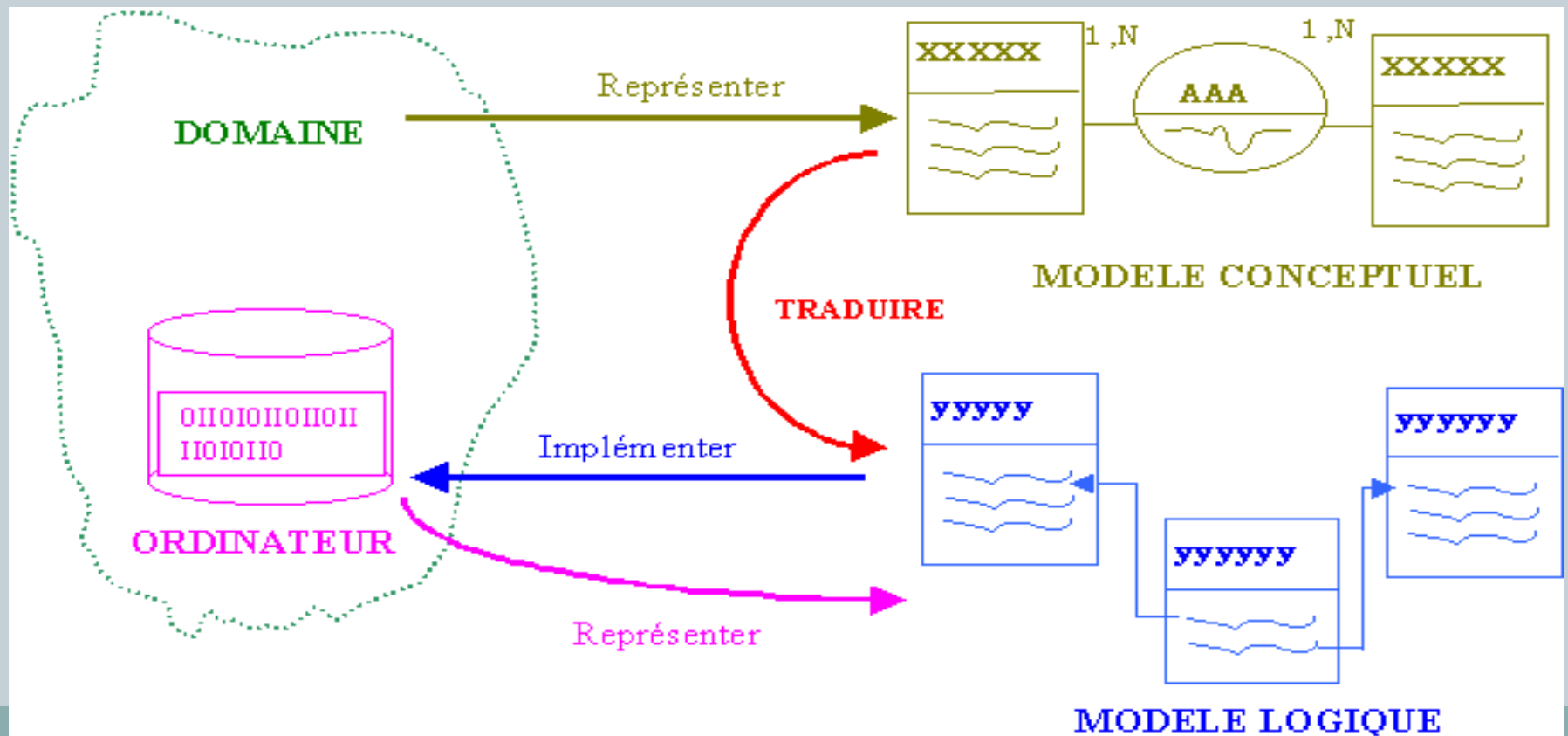
# Modèle Logique de Données



- **Un modèle** est un ensemble d'outils utilisés pour décrire et manipuler des données
- ✧ **Le modèle logique** est une REPRESENTATION du système tel qu'il sera implémenté dans des ordinateurs.
  - il ne faut pas confondre le modèle conceptuel (entité association par ex.) avec le modèle logique (relationnel par exemple)
  - il ne faut pas confondre le modèle logique (relationnel par ex. ) avec son implémentation physique en machine (avec Access ou Oracle par ex.)

# Modèle Logique de Données

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



# II/ Modèle relationnel

- ❧ Créé par CODD (IBM 69/70).
- La majorité des SGBD actuels sont basés sur ce modèle.
- Dispose d'un Langage d'interrogation et de requêtes simple
- **Principe simple** : 1 seul concept (relation ou table) pour décrire les données et les liens entre ses données.
- défini par la notion d'ensemble

## DERIVABILITY, REDUNDANCY AND CONSISTENCY OF RELATIONS STORED IN LARGE DATA BANKS

E. F. Codd  
Research Division  
San Jose, California

ABSTRACT: The large, integrated data banks of the future will contain many relations of various degrees in stored form. It will not be unusual for this set of stored relations to be redundant. Two types of redundancy are defined and discussed. One type may be employed to improve accessibility of certain kinds of information which happen to be in great demand. When either type of redundancy exists, those responsible for control of the data bank should know about it and have some means of detecting any "logical" inconsistencies in the total set of stored relations. Consistency checking might be helpful in tracking down unauthorized (and possibly fraudulent) changes in the data bank contents.

RJ 599 (# 12343)  
August 19, 1969

# MLD Relationnel



Rôle : préparer le modèle physique de données.

- Il s'appuie sur le modèle conceptuel de données.
- Les entités et certaines associations deviennent des relations (tables).
- Les propriétés deviennent des attributs (champs).
- Il fait apparaître les clés étrangères indispensables aux relations (liens) entre les données.



# Le modèle relationnel

## Concepts fondamentaux



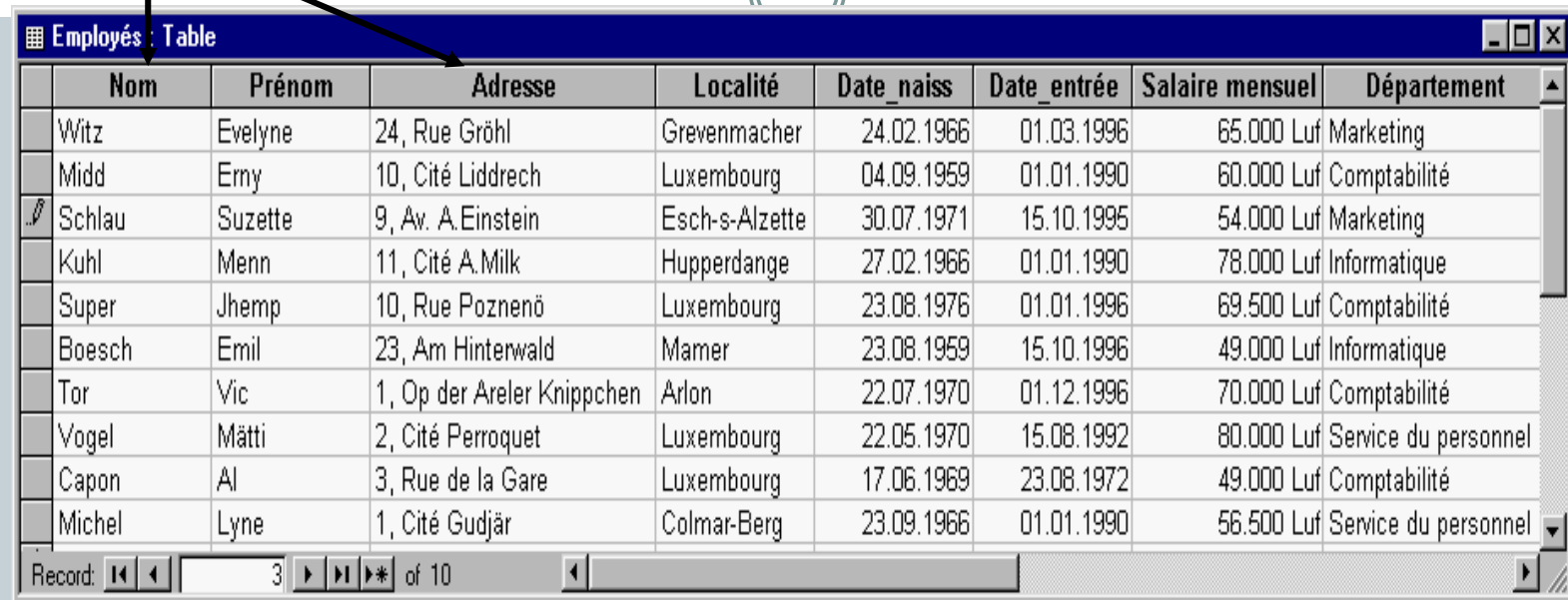
- **Relation**

- ⌘ sous-ensemble du produit cartésien de  $n$  domaines d'attribut ( $n > 0$ ).
- ⌘ caractérisée par:
  - un nom
  - des champs (attributs) qui correspondent aux noms de chacun des domaines mis en jeu (les colonnes)
  - des  $n$ -uplets (tuples) qui correspondent aux différentes combinaisons valides du produit cartésien (les lignes)

La relation est alors notée **R** (**A1:D1, A2:D2,..., An:Dn**)  
où **R** est le nom de la relation.

## ATTRIBUTS

# Concepts fondamentaux



	Nom	Prénom	Adresse	Localité	Date_naiss	Date_entrée	Salaire mensuel	Département
	Witz	Evelyne	24, Rue Gröhl	Grevenmacher	24.02.1966	01.03.1996	65.000 Luf	Marketing
	Midd	Erny	10, Cité Liddrech	Luxembourg	04.09.1959	01.01.1990	60.000 Luf	Comptabilité
	Schlau	Suzette	9, Av. A.Einstein	Esch-s-Alzette	30.07.1971	15.10.1995	54.000 Luf	Marketing
	Kuhl	Menn	11, Cité A.Milk	Hupperdange	27.02.1966	01.01.1990	78.000 Luf	Informatique
	Super	Jhemp	10, Rue Poznenö	Luxembourg	23.08.1976	01.01.1996	69.500 Luf	Comptabilité
	Boesch	Emil	23, Am Hinterwald	Mamer	23.08.1959	15.10.1996	49.000 Luf	Informatique
	Tor	Vic	1, Op der Areler Knippchen	Arlon	22.07.1970	01.12.1996	70.000 Luf	Comptabilité
	Vogel	Mätti	2, Cité Perroquet	Luxembourg	22.05.1970	15.08.1992	80.000 Luf	Service du personnel
	Capon	Al	3, Rue de la Gare	Luxembourg	17.06.1969	23.08.1972	49.000 Luf	Comptabilité
	Michel	Lyne	1, Cité Gudjär	Colmar-Berg	23.09.1966	01.01.1990	56.500 Luf	Service du personnel

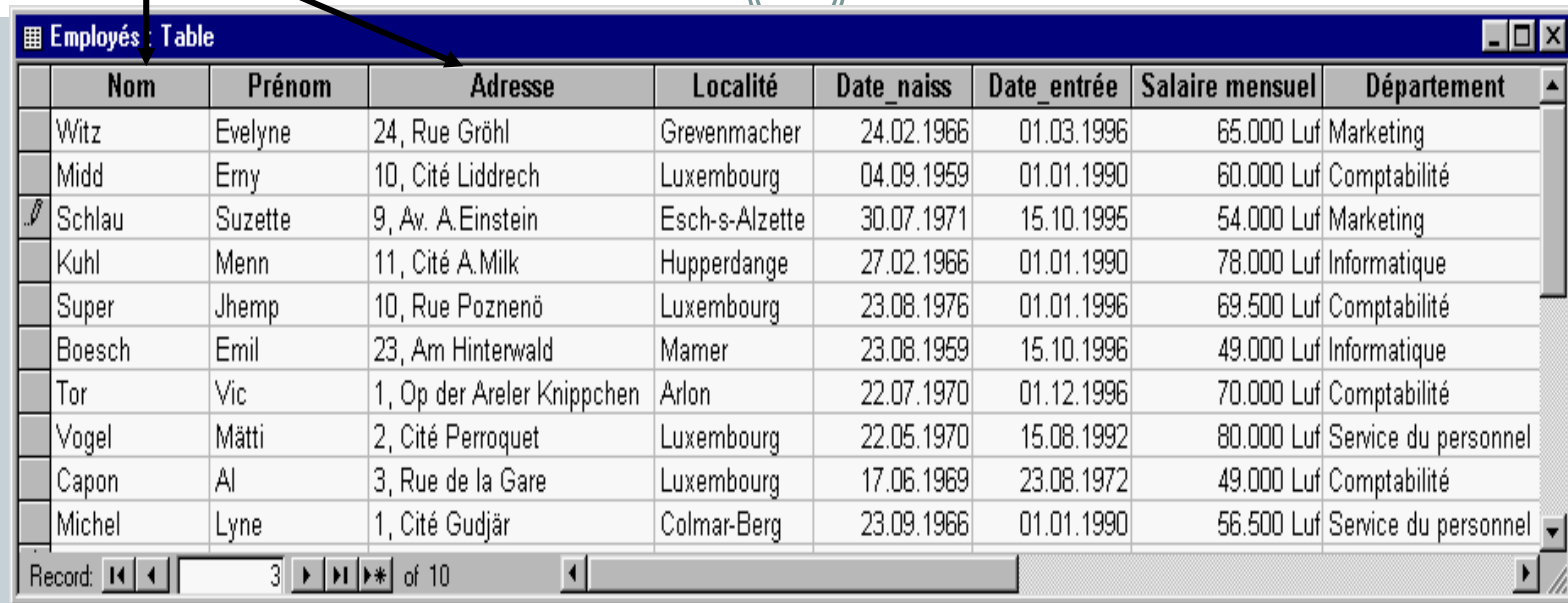
Record: 3 of 10

Exemple:

**EMPLOYE** (Nom: String, Prénom: String, Adresse: String, Localité: STRING, Date\_naiss: Date, Date\_entrée: Date, Salaire\_mensuel: Num, Département: String)

## ATTRIBUTS

# Concepts fondamentaux



	Nom	Prénom	Adresse	Localité	Date_naiss	Date_entrée	Salaire mensuel	Département
	Witz	Evelyne	24, Rue Gröhl	Grevenmacher	24.02.1966	01.03.1996	65.000 Luf	Marketing
	Midd	Erny	10, Cité Liddrech	Luxembourg	04.09.1959	01.01.1990	60.000 Luf	Comptabilité
	Schlau	Suzette	9, Av. A.Einstein	Esch-s-Alzette	30.07.1971	15.10.1995	54.000 Luf	Marketing
	Kuhl	Menn	11, Cité A.Milk	Hupperdange	27.02.1966	01.01.1990	78.000 Luf	Informatique
	Super	Jhemp	10, Rue Poznenö	Luxembourg	23.08.1976	01.01.1996	69.500 Luf	Comptabilité
	Boesch	Emil	23, Am Hinterwald	Mamer	23.08.1959	15.10.1996	49.000 Luf	Informatique
	Tor	Vic	1, Op der Areler Knippchen	Arlon	22.07.1970	01.12.1996	70.000 Luf	Comptabilité
	Vogel	Mätti	2, Cité Perroquet	Luxembourg	22.05.1970	15.08.1992	80.000 Luf	Service du personnel
	Capon	Al	3, Rue de la Gare	Luxembourg	17.06.1969	23.08.1972	49.000 Luf	Comptabilité
	Michel	Lyne	1, Cité Gudjär	Colmar-Berg	23.09.1966	01.01.1990	56.500 Luf	Service du personnel

Record: 3 of 10

**Tuple (ou *n*-uplet):** ligne de la table

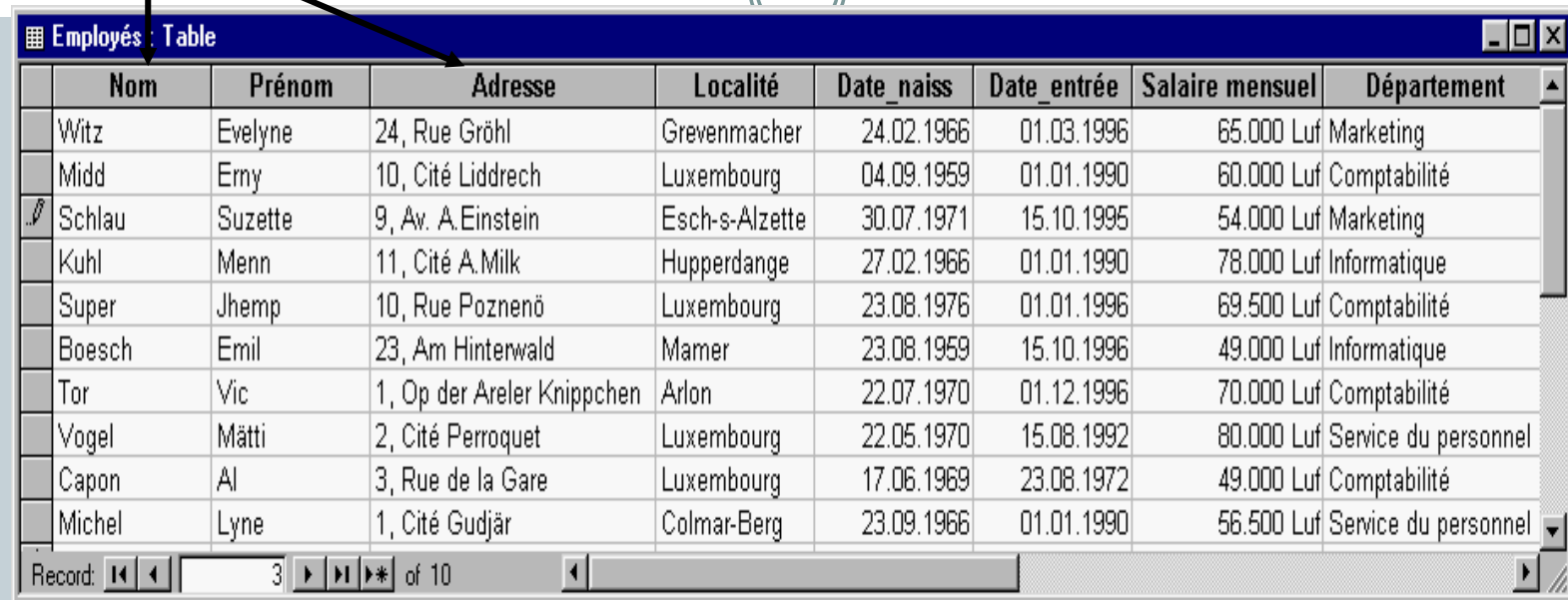
*Liste de n valeurs ( $v_1, \dots, v_n$ ) où chaque valeur  $v_i$  est la valeur d'un attribut  $A_i$  de domaine  $D_i$ .*

**Exemple :**

(‘Capon’, ‘Al’, ‘3, Rue de la Gare’, ‘Luxembourg’, 17.06.1969, 23.08.1972, 49.000, ‘comptabilité’)

## ATTRIBUTS

# Concepts fondamentaux



	Nom	Prénom	Adresse	Localité	Date_naiss	Date_entrée	Salaire mensuel	Département
	Witz	Evelyne	24, Rue Gröhl	Grevenmacher	24.02.1966	01.03.1996	65.000 Luf	Marketing
	Midd	Erny	10, Cité Liddrech	Luxembourg	04.09.1959	01.01.1990	60.000 Luf	Comptabilité
	Schlau	Suzette	9, Av. A.Einstein	Esch-s-Alzette	30.07.1971	15.10.1995	54.000 Luf	Marketing
	Kuhl	Menn	11, Cité A.Milk	Hupperdange	27.02.1966	01.01.1990	78.000 Luf	Informatique
	Super	Jhemp	10, Rue Poznenö	Luxembourg	23.08.1976	01.01.1996	69.500 Luf	Comptabilité
	Boesch	Emil	23, Am Hinterwald	Mamer	23.08.1959	15.10.1996	49.000 Luf	Informatique
	Tor	Vic	1, Op der Areler Knippchen	Arlon	22.07.1970	01.12.1996	70.000 Luf	Comptabilité
	Vogel	Mätti	2, Cité Perroquet	Luxembourg	22.05.1970	15.08.1992	80.000 Luf	Service du personnel
	Capon	Al	3, Rue de la Gare	Luxembourg	17.06.1969	23.08.1972	49.000 Luf	Comptabilité
	Michel	Lyne	1, Cité Gudjär	Colmar-Berg	23.09.1966	01.01.1990	56.500 Luf	Service du personnel

Record: 3 of 10

**Combien y a-t-il de tuple dans la table EMPLOYÉS?**

# Concepts fondamentaux



## ∞ Domaine

Un domaine est un ensemble de valeurs que peut prendre un attribut; c'est le domaine définition d'un ou plusieurs attributs.

Exemple de domaines:

- $D_{\text{nom}}$  : chaînes de caractères de longueur maximale 30
- $D_{\text{num}}$  : entiers compris entre 0 et 99999
- $D_{\text{couleur}}$  : {"bleu", "vert", "jaune"}
- $D_{\text{âge}}$  : entiers compris entre 16 et 65

# Concepts fondamentaux



- **Attribut**

Nomment les colonnes d'une relation.

Un attribut est toujours associé à un domaine.

# Concepts fondamentaux



Employés : Table								
	Nom	Prénom	Adresse	Localité	Date_naiss	Date_entrée	Salaire mensuel	Département
	Witz	Evelyne	24, Rue Gröhl	Grevenmacher	24.02.1966	01.03.1996	65.000 Luf	Marketing
	Midd	Erny	10, Cité Liddrech	Luxembourg	04.09.1959	01.01.1990	60.000 Luf	Comptabilité
	Schlau	Suzette	9, Av. A.Einstein	Esch-s-Alzette	30.07.1971	15.10.1995	54.000 Luf	Marketing
	Kuhl	Menn	11, Cité A.Milk	Hupperdange	27.02.1966	01.01.1990	78.000 Luf	Informatique
	Super	Jhemp	10, Rue Poznenö	Luxembourg	23.08.1976	01.01.1996	69.500 Luf	Comptabilité
	Boesch	Emil	23, Am Hinterwald	Mamer	23.08.1959	15.10.1996	49.000 Luf	Informatique
	Tor	Vic	1, Op der Areler Knippchen	Arlon	22.07.1970	01.12.1996	70.000 Luf	Comptabilité
	Vogel	Mätti	2, Cité Perroquet	Luxembourg	22.05.1970	15.08.1992	80.000 Luf	Service du personnel
	Capon	Al	3, Rue de la Gare	Luxembourg	17.06.1969	23.08.1972	49.000 Luf	Comptabilité
	Michel	Lyne	1, Cité Gudjâr	Colmar-Berg	23.09.1966	01.01.1990	56.500 Luf	Service du personnel
Record: 3 of 10								

Exemple:

Attributs: Nom, Prénom, Adresse, Localité, Date\_naiss, Date\_entrée, Salaire mensuel, Département

# Exemple : une relation



	Nom	Prénom	Adresse	Localité	Date_naiss	Date_entrée	Salaire mensuel	Département
	Witz	Evelyne	24, Rue Gröhl	Grevenmacher	24.02.1966	01.03.1996	65.000 Luf	Marketing
	Midd	Erny	10, Cité Liddrech	Luxembourg	04.09.1959	01.01.1990	60.000 Luf	Comptabilité
	Schlau	Suzette	9, Av. A.Einstein	Esch-s-Alzette	30.07.1971	15.10.1995	54.000 Luf	Marketing
	Kuhl	Menn	11, Cité A.Milk	Hupperdange	27.02.1966	01.01.1990	78.000 Luf	Informatique
	Super	Jhemp	10, Rue Poznenö	Luxembourg	23.08.1976	01.01.1996	69.500 Luf	Comptabilité
	Boesch	Emil	23, Am Hinterwald	Mamer	23.08.1959	15.10.1996	49.000 Luf	Informatique
	Tor	Vic	1, Op der Areler Knippchen	Arlon	22.07.1970	01.12.1996	70.000 Luf	Comptabilité
	Vogel	Mätti	2, Cité Perroquet	Luxembourg	22.05.1970	15.08.1992	80.000 Luf	Service du personnel
	Capon	Al	3, Rue de la Gare	Luxembourg	17.06.1969	23.08.1972	49.000 Luf	Comptabilité
	Michel	Lyne	1, Cité Gudjâr	Colmar-Berg	23.09.1966	01.01.1990	56.500 Luf	Service du personnel

Record: 3 of 10

- Relation (table): employé
- Attributs: Nom, Prénom, Adresse, Localité, Date\_naiss, Date\_entrée, Salaire mensuel, Département
- 10 tuples (les lignes)



# Concepts fondamentaux



## Degré d'une relation:

- DEGRÉ (ou encore, ORDRE, ARITÉ) d'une relation : le nombre des attributs (ou colonnes) qu'elle contient.

Rien à voir donc avec la dimension d'une association.

# Concepts fondamentaux



## **Caractéristiques d'une relation:**

- L'ordre des lignes et des colonnes n'est pas significatif
- Pas de lignes identiques (comment?)
- Une case ne peut contenir qu'une valeur au plus (une case est l'intersection d'une ligne et d'une colonne)?

# Concepts fondamentaux



- **Base de données relationnelle:**

BDD dont le schéma est un ensemble de schémas de relations et dont les occurrences sont les tuples de ces relations.

- **Système de gestion de bases de données relationnel (SGBDR) :**

C'est un logiciel supportant le modèle relationnel, et qui peut manipuler les données avec des opérateurs relationnels.

# clé primaire



Pour identifier chaque ligne dans une relation, la relation doit avoir une clé.

## ***Définition:***

*Une clé est un groupe d'attributs minimal qui détermine un tuple d'une manière unique dans une relation.*

# clé primaire



## **Exemples de clés :**

- le Numéro d'identification à la sécurité sociale
- Le Numéro (Matricule) de l'étudiant dans un service de scolarité
- Le Numéro d'adhérent dans la bibliothèque nationale

# clé primaire



## **Un autre exemple de clé:**

Dans les bibliothèques, on utilise pour la table OUVRAGES une clé qui est l'attribut « cote » de la table.

La cote permet de déterminer de façon unique une ligne de la table

# clé primaire



- ✧ Une relation doit avoir une clé primaire
- ✧ Une clé primaire peut être:
  - Soit simple (constituée d'1 seul attribut)
  - Soit composée (constituée de plusieurs attributs)
- ✧ Clé primaire => Irréductibilité + unicité+sa valeur ne peut être nulle

# clé primaire



## **Clé candidate:**

Un attribut pouvant jouer le rôle d'une clé primaire dans une relation.

Exemple:

Discuter des clés candidates dans la table EMPLOYÉS?



# clé primaire



## ∞ Clé étrangère:

un attribut dans une relation, qui est en même temps la clé primaire d'une autre relation

⇒ **Clé référentielle** (elle permet de faire le lien entre relations)

# clé primaire



## Autres exemples:

- **EMPLOYE (matricule, nom, salaire)** a 1 clé candidate : matricule

Si on est certain que 2 employés n'ont jamais le même **nom**

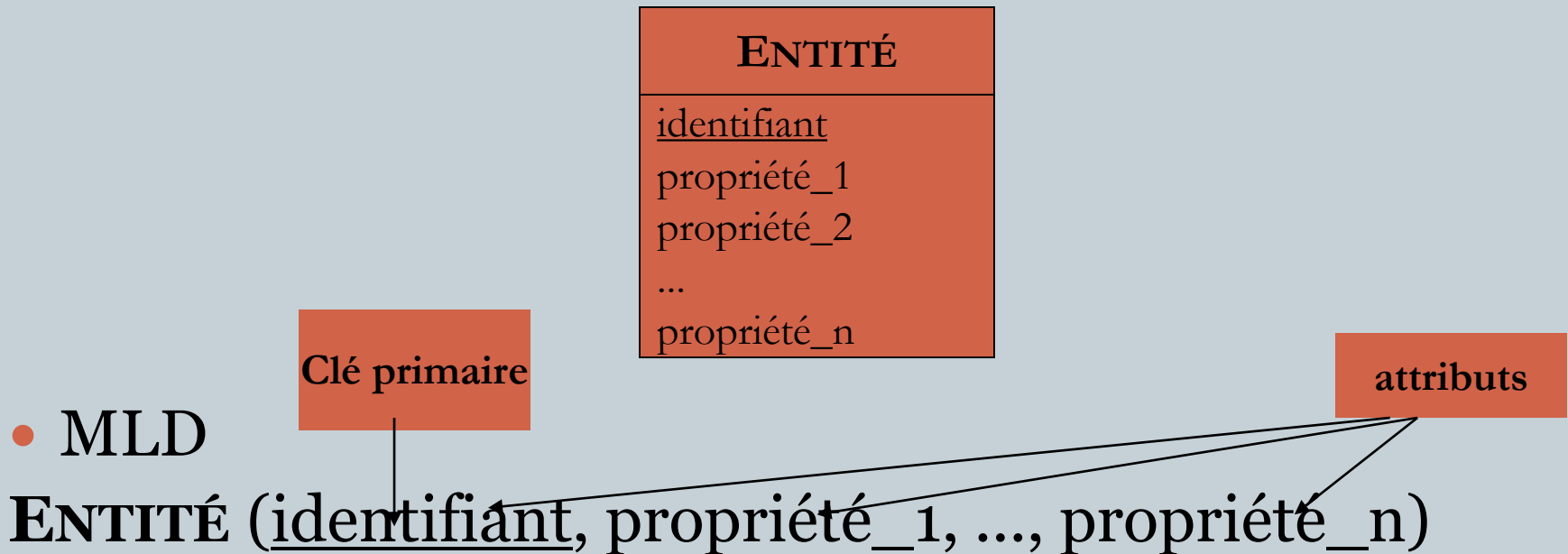
**nom** est une 2ème clé candidate.

- **EMPLOYES(matricule, nom, prénom, salaire)** a 2 clés candidates :
  - **matricule** (que l'on choisira certainement comme clé primaire)
  - **(nom, prénom)**, En supposant que 2 employés n'ont jamais les mêmes noms et prénoms.

# III/ Passage du MCD vers MLDR



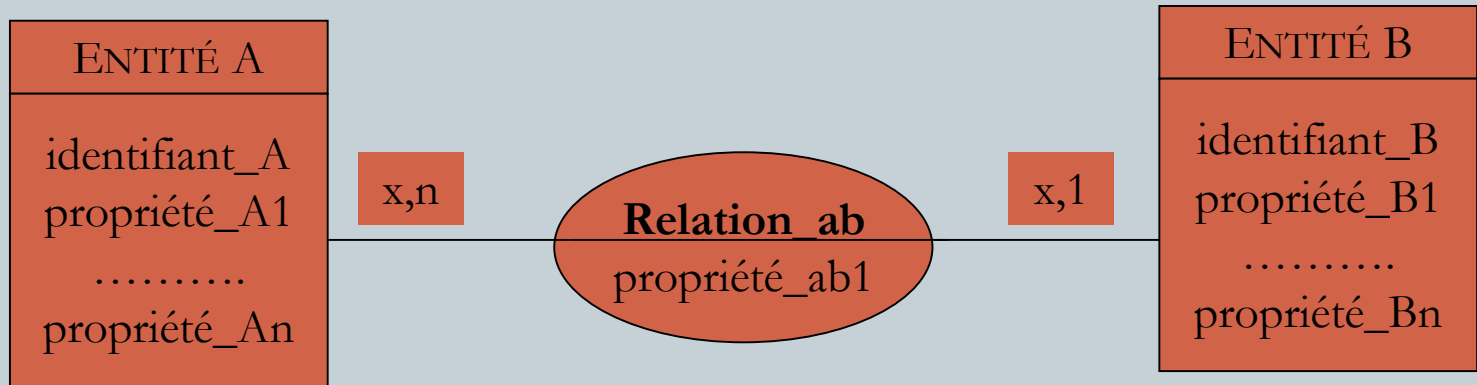
- Une entité se transforme en relation (table)



# Relation binaire aux cardinalités $x, n-x, 1$ ( $x=0$ ou $1$ )



La clé primaire de la relation à la cardinalité  $x, n$  devient clé étrangère dans la table à la cardinalité  $x, 1$



MLD :

**ENTITÉ A** (identifiant A, propriété\_A1, ..., propriété\_An)

**ENTITÉ B** (identifiant B, propriété\_B1, ..., propriété\_Bn, #identifiant\_A, propriété\_ab1)



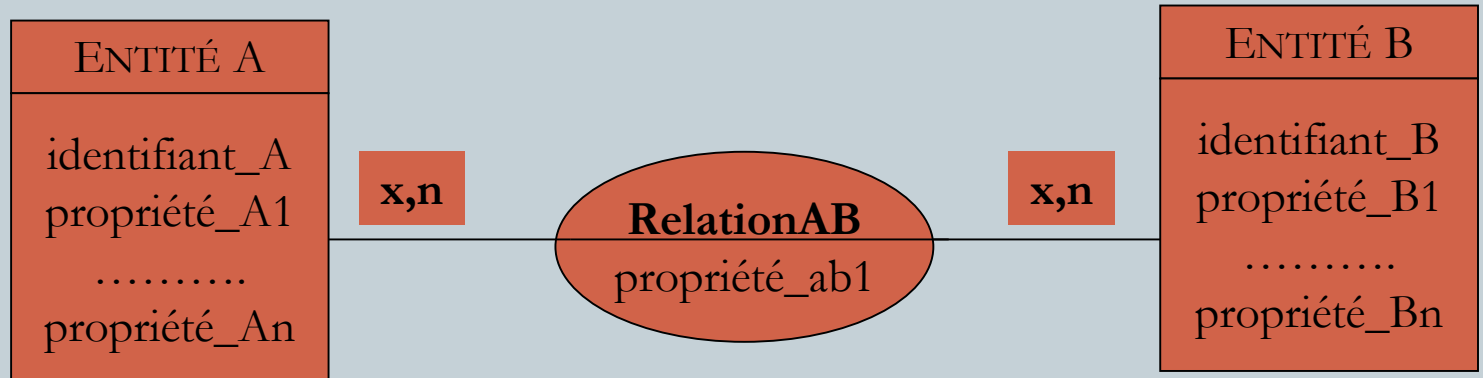
Clé étranger

Propriété association

# Relation binaire aux cardinalités $x, n-x, n$ ( $x=0$ ou $1$ )



création d'une relation supplémentaire ayant comme clé primaire une clé composée des identifiants des deux entités.



**MLD :**

**ENTITÉ A** (identifiant A, propriété\_A1, ..., propriété\_An)

**ENTITÉ B** (identifiant B, propriété\_B1, ..., propriété\_Bn)

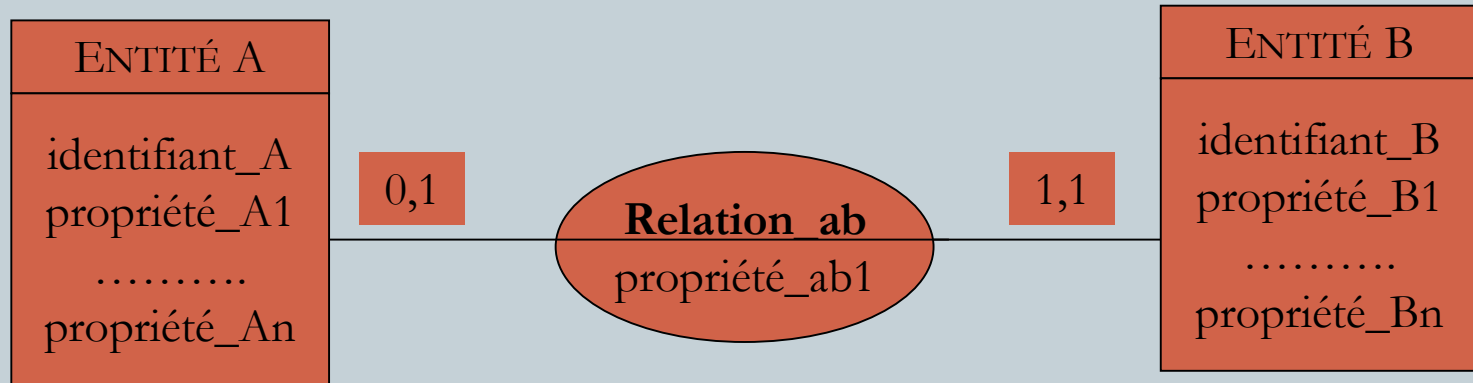
**RELATION AB** (#identifiant A, #identifiant B,  
propriété\_ab1)



Clé multi attributs

# Relation binaire aux cardinalités 0,1-1,1

- La clé primaire de la relation à la cardinalité 0,1 devient clé étrangère dans la relation à la cardinalité 1,1.



MLD :

**ENTITÉ A** (identifiant\_A, propriété\_A1, ..., propriété\_An)

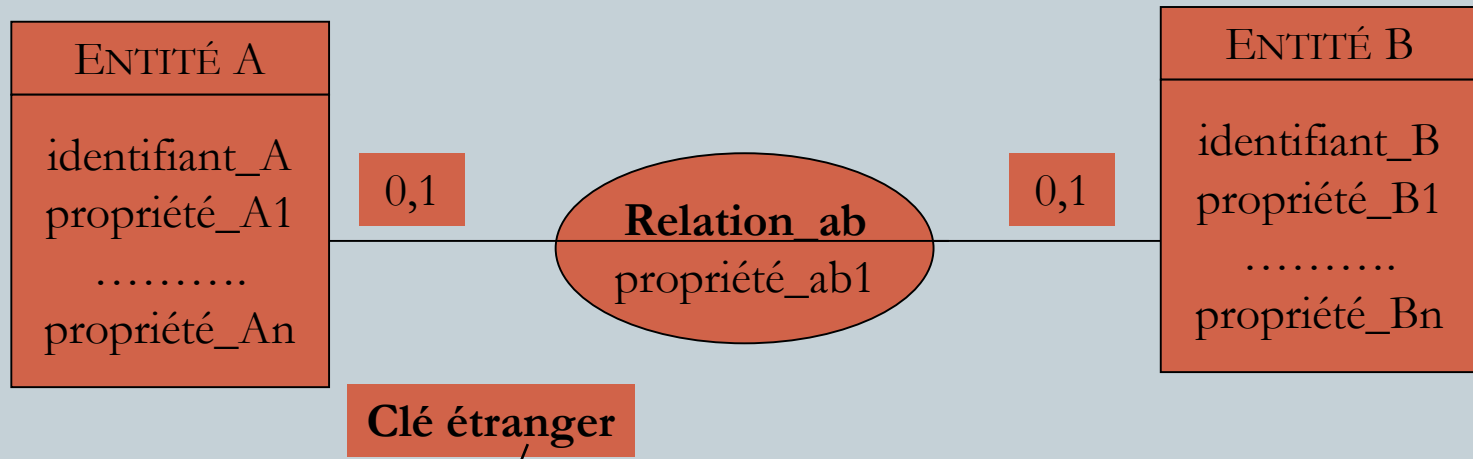
**ENTITÉ B** (identifiant\_B, propriété\_B1, ..., propriété\_Bn,  
#identifiant\_A, propriété\_ab1)



Clé étrangère

# Relation binaire aux cardinalités 0,1-0,1

La clé primaire d'une des relations devient clé étrangère de l'autre.

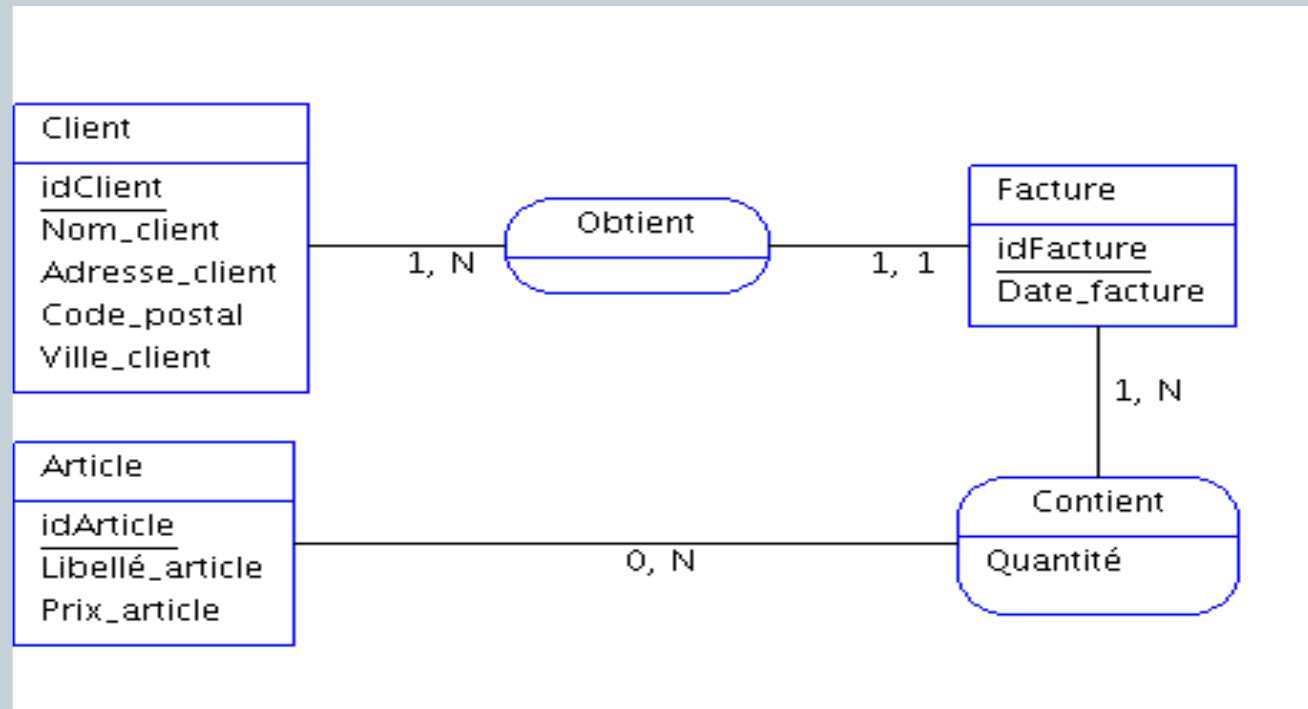


MLD :

- **ENTITÉ A** (`identifiant_A`, `propriété_A1`, ..., `propriété_An`, `#identifiant_B`, **`propriété_ab1`**)
- **ENTITÉ B** (`identifiant_B`, `propriété_B1`, ..., `propriété_Bn`)

# Exemple 1

## soit le MCD



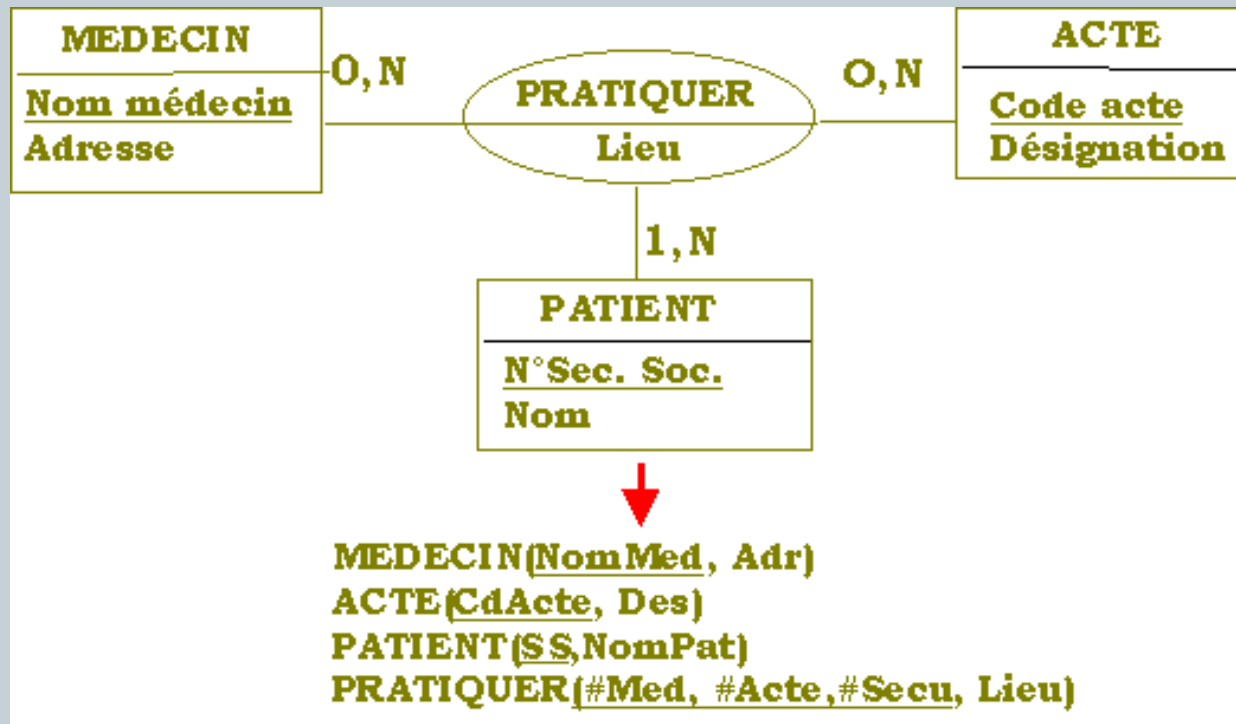


# MLDR associé

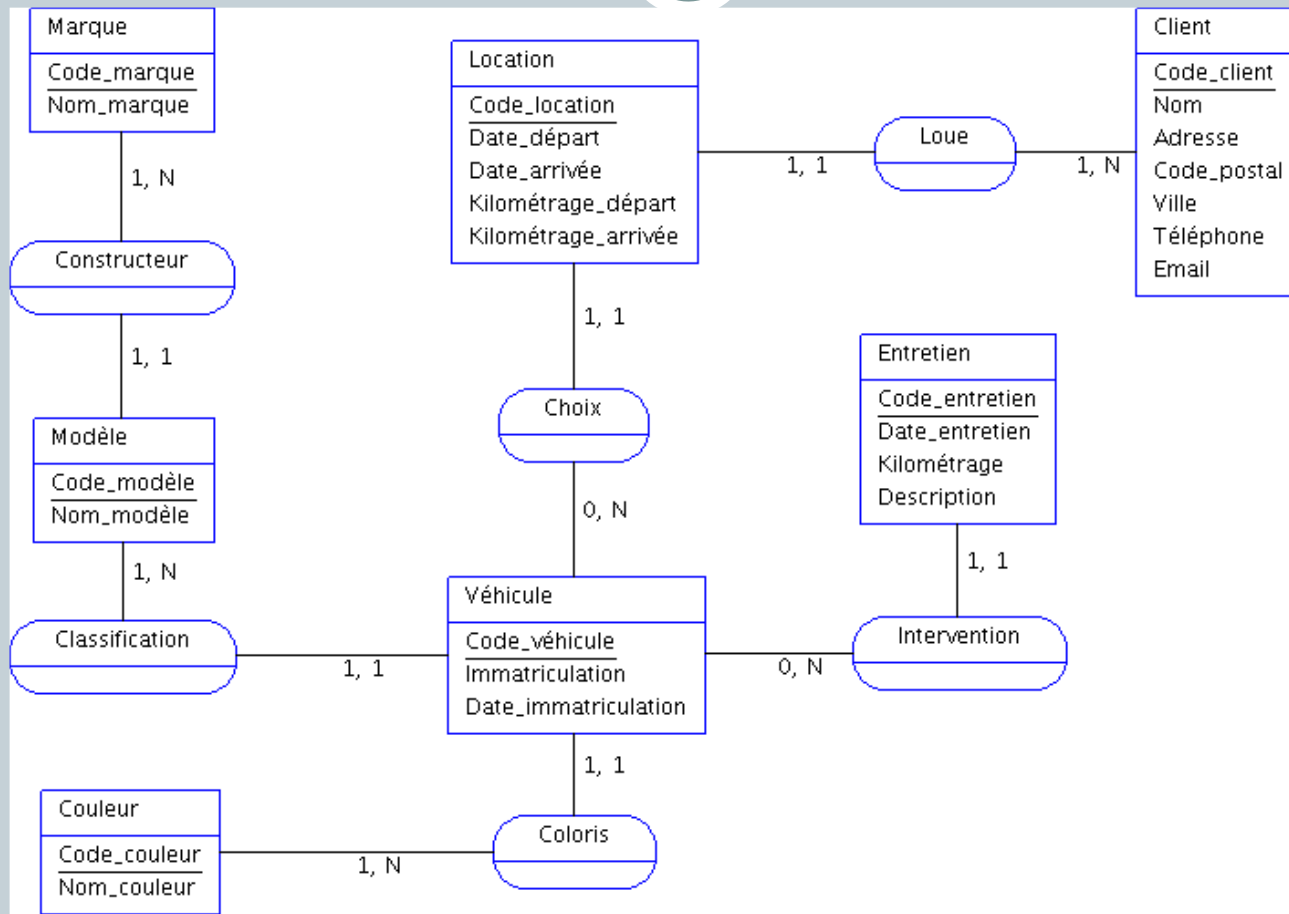


- **Client(idClient, nom\_client, adresse\_client, code\_postal, ville\_client)**
- **Facture(idFacture, date\_facture, #idClient)**
- **Contient(#idFacture, #idArticle, quantité)**
- **Article(idArticle, libelle\_article, prix\_article)**

# Exemple2: Passage MCD vers MLD



# Exemple3: Passage MCD vers MLD



# Exemple3: MLDR



- **Location** (Code location, Date\_départ, Date\_arrivée, Kilométrage\_départ, Kilométrage\_arrivée, #Code\_client, #Code\_véhicule)
- **Véhicule** (Code véhicule, Immatriculation, Date\_immatriculation, #Code\_couleur, #Code\_modèle)
- **Client** (Code client, Nom, Adresse, Code\_postal, Ville, Téléphone, Email)
- **Modèle** (Code modèle, Nom\_modèle, #Code\_marque)
- **Marque** (Code marque, Nom\_marque)
- **Couleur** (Code couleur, Nom\_couleur)
- **Entretien** (Code entretien, Date\_entretien, Kilométrage, Description, #Code\_véhicule)

# IV / Contraintes d'intégrité



- une assertion (ou condition) qui doit être vrai durant toute la vie de la base de donnée
- Permet de limiter les possibilités pour les valeur affectées aux champs dans les tuples (les valeurs d'une propriété)
- Les contraintes d'intégrité sont vérifiées (exécutées) à chaque mise à jour de la base de données
- Une base de donnée est dite cohérente si toutes les contraintes d'intégrité sont vérifiées

# IV / Contraintes d'intégrité



## **Type de contrainte d'intégrité:**

- *Intégrité de clé*
- *Intégrité de domaine (type)*
- *Intégrité de référence*

# Contraintes d'intégrité



## **CONSTRAINTES DE CLÉ**

Une relation doit posséder une clé primaire;  
Une clé primaire doit être unique et non nulle;

# Contraintes d'intégrité



## **CONSTRAINTES DE DOMAINE**

Tout attribut de la relation appartient un domaine.  
Un domaine peut être fini ou non fini.



# Contraintes d'intégrité



## CONTRAINTES DE RÉFÉRENCE

Cette contrainte concerne les clés étrangères; c'est une contrainte exprimée entre deux tables.

Tout tuple d'une relation faisant référence à une autre relation doit se référer à un tuple qui existe déjà; autrement dit, le tuple référencé doit avoir été saisi dans la table référencée avant même de s'y référer.

# Contraintes d'intégrité pour un attribut: possibilités



Contrainte	Effet de la contrainte
Être non NULL	la valeur de l'attribut ne peut être vide.
Être unique	la valeur de l'attribut pour une ligne est unique (clé)
Être fini	contenir uniquement certaines valeurs (fini). <u>Exemple</u> : Couleur={Rouge,Vert,Bleu}. Sexe = {féminin; masculin}
Être bornée	<u>Exemple</u> : Date doit être supérieure au 01/01/1900 et inférieure au 31/12/2040.
Être calibrée	<u>Exemple</u> : Téléphone doit se présenter selon la norme OXXX XX XX XX.

# Contraintes d'intégrité



## **Exemple :**

la relation MARQUE

**MARQUE (IdMarque, NomMarque, ClasseMarque, IdFabriquant)**

## **Domaines :**

Dom (IdMarque) = [1..99 999]

Dom (NomMarque) = ensemble de tous les mots construits sur l'alphabet {A, B, ..Z, a, b, .., z, 0.. 9}

Dom (ClasseMarque) = [1..30]

Dom (IdFabriquant) = tous les noms possibles de sociétés (chaînes limitées à 100 caractères)

# Contraintes d'intégrités



## **Exemple :**

Soit le schéma d'une base de données relationnelle permettant la gestion de notices bibliographiques :

- EDITEUR (**NumEditeur**, Nom, Prénom, adresse, tel)
- AUTEURS (**NumAuteur**, Nom, Prénom)
- OUVRAGES (**Cote**, Titre, NbExemplaire, Année, #NumEditeur, Thème)

On souhaite poser les contraintes suivantes :

1. Le nombre d'exemplaires de chaque OUVRAGE doit être supérieur à 0 (zéro) et inférieur à 50.
2. Chaque OUVRAGE doit avoir au moins un auteur.

**Questions :** Pour chacun des deux cas :

1. Donnez le type de (des) la contrainte(s) d'intégrité impliqué(es).
2. Écrire la contrainte.

# Contraintes d'intégrité



Le nombre d'exemplaires de chaque OUVRAGE doit être supérieur à 0 (zéro) et inférieur à 50.

**Type de contrainte :** CI de domaine

**Attribut concerné :** NbExemplaire (relation : OUVRAGES)

**Écriture de contrainte:**  $(\text{NbExemplaire} > 0)$  et  $(\text{NbExemplaire} < 50)$

# Contraintes d'intégrité



Chaque OUVRAGE doit avoir au moins un auteur.

## **Analyse :**

Il ne peut pas s'agir d'une CI de domaine (sinon, quel est l'attribut concerné?)

Il ne peut pas s'agir non plus d'une CI de clé puisqu'aucun attribut n'est cité dans la contrainte.

Problème : Dans le schéma de la BD, il n'y a aucune modélisation des données citées dans la contrainte. En d'autres termes, il n'y a aucun lien entre les deux relations OUVRAGES et AUTEURS qui permet de répondre aux deux questions : Quel(s) auteur(s) a écrit quel(s) ouvrage(s)? Quel(s) ouvrage(s) a (ont) été écrit(s) par quel(s) auteur(s)?

Toutes ces données, il faut les modéliser dans la BD.

# Contraintes d'intégrité



## Modélisation:

Un ouvrage est écrit par un ou plusieurs auteurs.

Un auteurs peut écrire un ou plusieurs ouvrages.

Ceci peut être modélisé par la création d'une nouvelle relation ÉCRIT  
faisant référence:

et à l'auteur

et aux ouvrages qu'il a écrit

ou autrement:

et à l'ouvrage

et aux auteurs qui l'ont écrit.

# Contraintes d'intégrité



Schéma en intension : ÉCRIT (#NumAuteur, #cote)

Schéma en extension :

Relation : ÉCRIT

<u>NumAuteur</u>	<u>cote</u>
12	45
12	55
88	45
29	3
12	3



# Contraintes d'intégrité



Maintenant, il faut écrire les CI associées à ce schéma:

## **CI DE CLÉ PRIMAIRE :**

NumAuteur, cote (unique et non nulle)

## **CI DE CLÉ RÉFÉRENTIELLE :**

ECRIT.NumAuteur      **REFERENCE AUTEURS.NumAuteur**

ECRIT.cote              **REFERENCE OUVRAGES.cote**