



# MODÉLISATION DE BASES DE DONNÉES RELATIONNELLES

Mouna Kamel  
L2 Info– 2025/2026



# Plan du cours

- Introduction
  - *Un exemple*
  - *Bases de Données : définitions*
- Dépendances fonctionnelles
- Modèle Entités/Associations (MEA)
- Modèle Relationnel
  - Définitions
  - Règles de Traduction d'un MEA en Modèle Relationnel
- Normalisation
- Héritage

# Plan du cours

- Introduction
  - Un exemple
  - Bases de Données : définitions
- Dépendances fonctionnelles
- Modèle Entités/Associations (MEA)
- Modèle Relationnel
  - Définitions
  - Règles de Traduction d'un MEA en Modèle Relationnel
- Normalisation
- Héritage

# Modèle Entités/Associations

- Modèle conceptuel des données conçu par P. Chen dans les années 70
- Description graphique pour représenter des modèles de données sous la forme de diagramme contenant des entités et des associations.

# Modèle Entités/Associations

## ■ Notions de base

- **Propriété** : information élémentaire (atomique) : *référence* d'un article, *prix* d'un article, *code* du client
- **Entité** : objet reflet d'une entité dotée d'une existence identifiable, qui regroupe des propriétés : objets *Article*, *Client*
- **Association / Relation**: représentation d'une association entre plusieurs entités: relation *commander* entre les objets *Client* et *Article*
- **Cardinalité** : indique pour chaque couple (objet, relation) les nombres minimum et maximum de valeurs de la relation pouvant exister pour chaque valeur de l'objet : (*Client*, *commander*) a pour cardinalité 0-n Les valeurs possibles des cardinalités sont : (0,1), (1,1); (0,N), (1,N)
- **Contrainte d'Intégrité** : indique une dépendance obligatoire entre plusieurs objets participant à une relation : un *Article* ne peut être fourni que par un seul *Fournisseur*

# Modèle Entités/Associations

**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé ne peut travailler que dans un seul service

On identifie 2 entités : **Employé** et **Service**

Propriétés de chaque entité :

**Employé** : num\_emp, Nom\_emp, Date\_naiss

**Service** : Num\_serv, Nom\_serv, Bâtiment

Analyse des DF :

Num\_emp --> Nom\_emp, Date\_naiss, Num\_serv, Nom\_serv, bâtiment

Num\_serv --> Nom\_serv, bâtiment

# Modèle Entités/Associations

**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé ne peut travailler que dans un seul service

On identifie 2 entités : **Employé** et **Service**

Propriétés de chaque entité :

**Employé** : num\_emp, Nom\_emp, Date\_naiss

**Service** : Num\_serv, Nom\_serv, Bâtiment

Analyse des DF équivalentes en utilisant la propriété de transitivité :

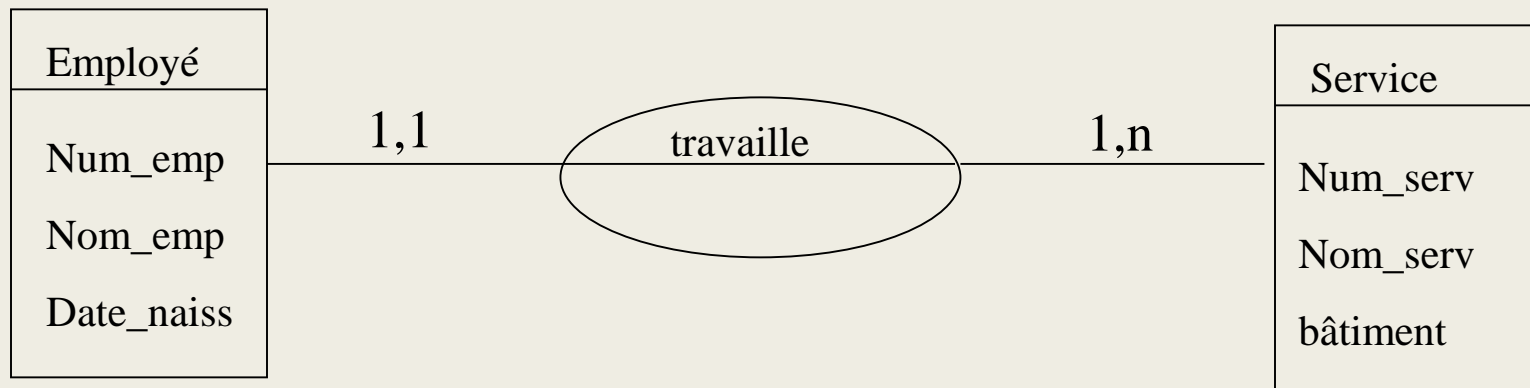
Num\_emp --> Nom\_emp, Date\_naiss, Num\_serv

Num\_serv --> Nom\_serv, bâtiment

# Modèle Entités/Associations

**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé ne peut travailler que dans un seul service





# Modèle Entités/Associations

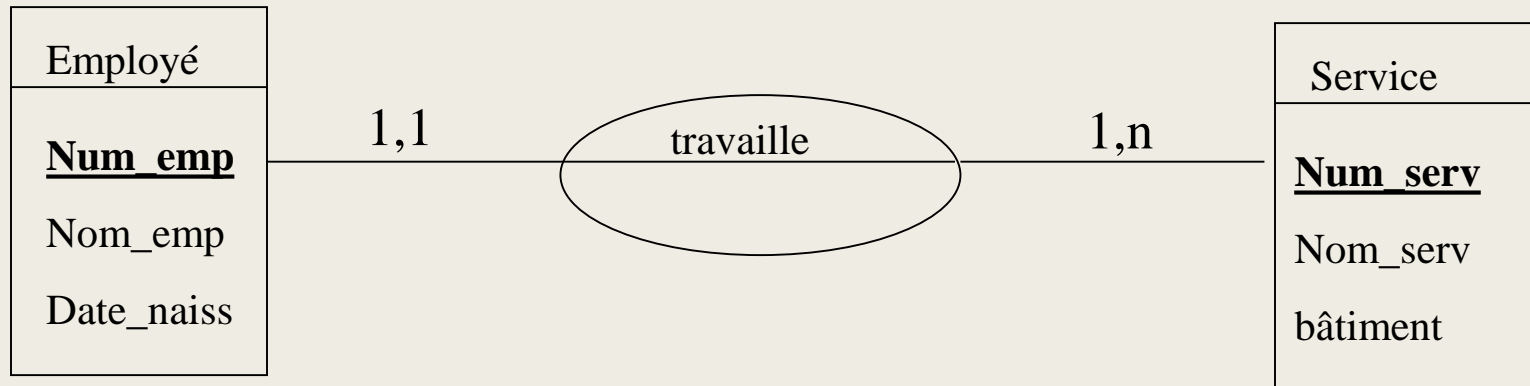
- Une **clé** est un ensemble minimum d'attributs dont la connaissance des valeurs permet d'identifier une seule occurrence de l'entité.

Exemples de clé : num\_emp, num\_serv, etc.,  
mais aussi [n° salle, dateProjection, heureDebut], etc.

# Modèle Entités/Associations

**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé ne peut travailler que dans un seul service



# Modèle Entités/Associations

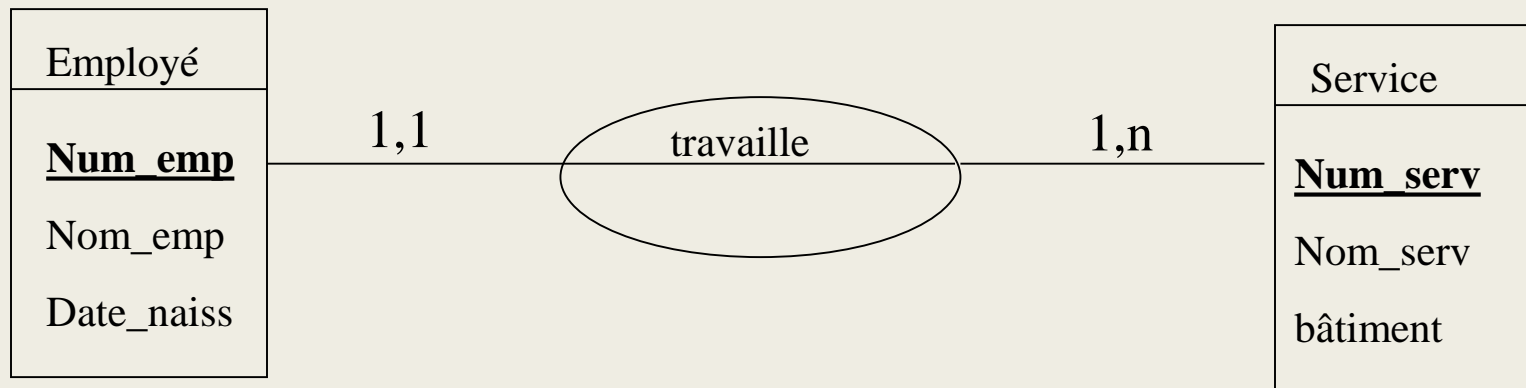
## Exercices

Exercices 7, 8 et 9

# Modèle Entités/Associations

**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé ne peut travailler que dans un seul service



- 1) Tous les attributs sont atomiques
- 2) Définir la clé au sein de chaque Entité : la clé d'une Entité doit être en DF avec tous les autres attributs de l'Entité
- 3) Etablir les cardinalités des relations
- 4) Une propriété ne doit apparaître qu'une seule fois dans le modèle

# Modèle Entités/Associations

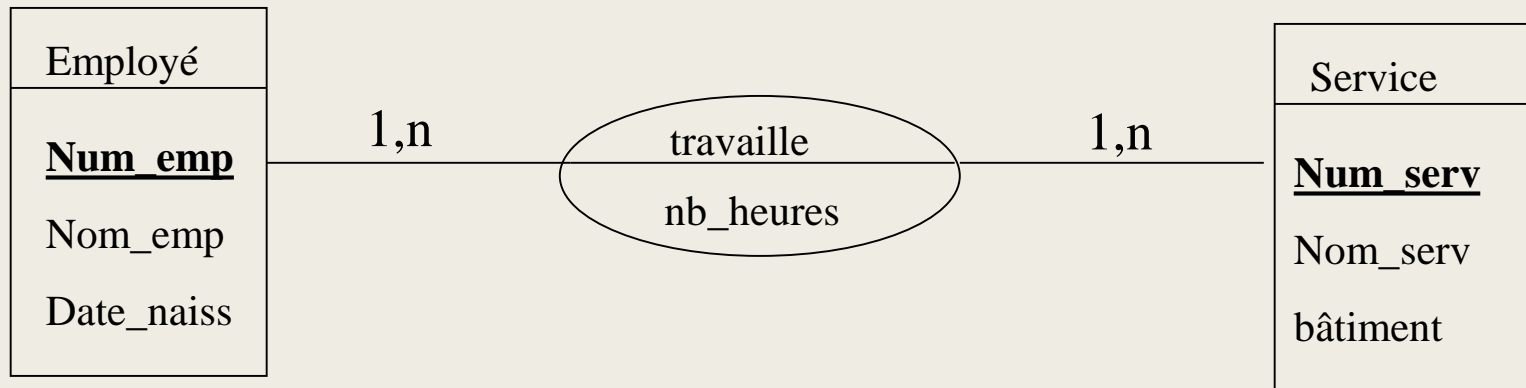
**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé peut travailler dans plusieurs services, et consacre un certain nombre d'heures hebdomadaires à chaque service

Num\_emp --> Nom\_emp, Date\_naiss

Nul\_serv --> Nom\_serv, bâtiment

num\_emp, num\_service --> nb\_heures



→ Tous les attributs qui dépendent fonctionnellement des clés des entités se mettent dans l'association qui relie ces entités

# Modèle Entités/Associations

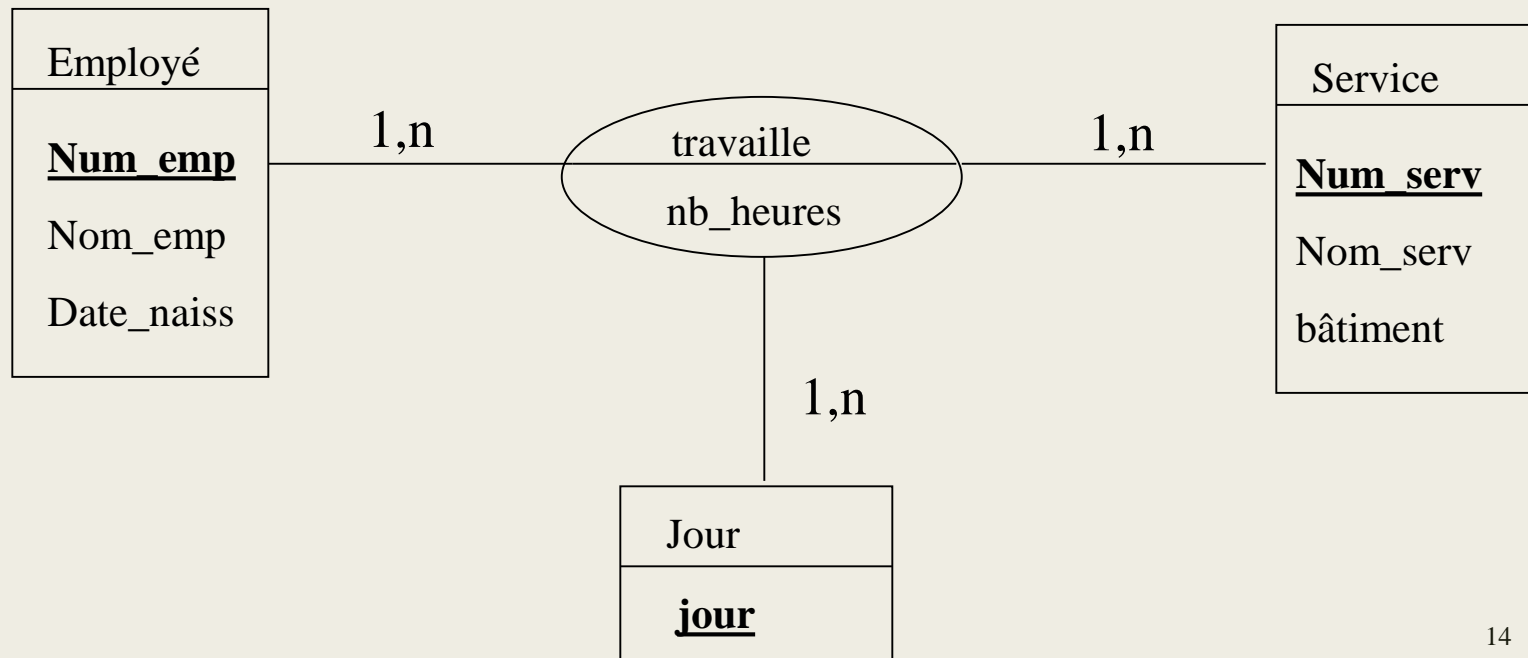
**Univers** : employés d'une entreprise composée de différents services

**Règle de gestion** : un employé peut travailler dans plusieurs services, et consacre un certain nombre d'heures par jour à chaque service

Num\_emp --> Nom\_emp, Date\_naiss

Num\_serv --> Nom\_serv, bâtiment

num\_emp, num\_service, jour --> nb\_heures



# Modèle Entités/Associations

## Exercices

Exercices 10, 11, 12 et 13

# Modèle Entités/Associations

- CIF : Contrainte d'Intégrité Fonctionnelle

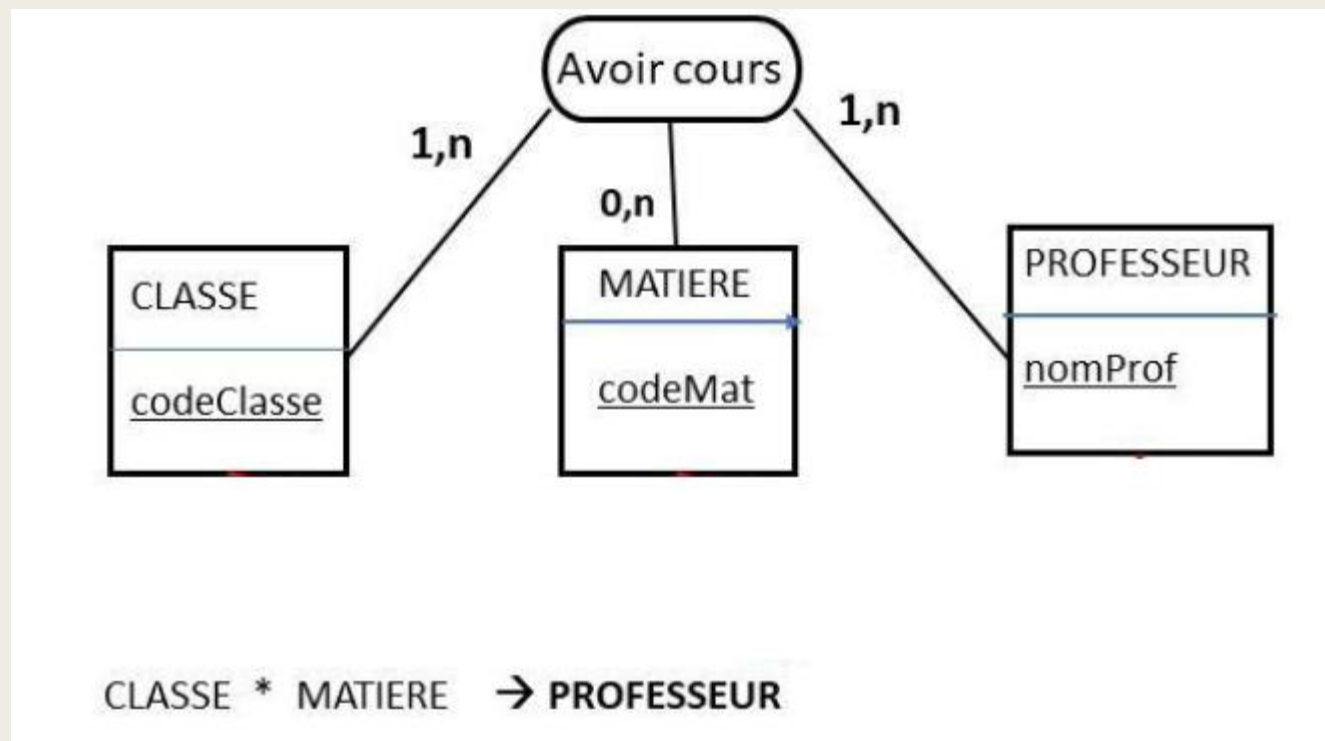
Dans une relation de dimension  $> 2$  (ternaire ou plus), la notion de dépendance fonctionnelle peut impliquer tout ou partie des entités et n'est pas systématiquement liée aux cardinalités.

Elle a donc une modélisation explicite et prend alors le nom de contrainte d'intégrité fonctionnelle, représentée par un rond noté CIF, relié à la relation porteuse (trait pointillé) et aux entités concernées (trait plein) dont l'une est porteur d'une flèche (la cible de la CIF).



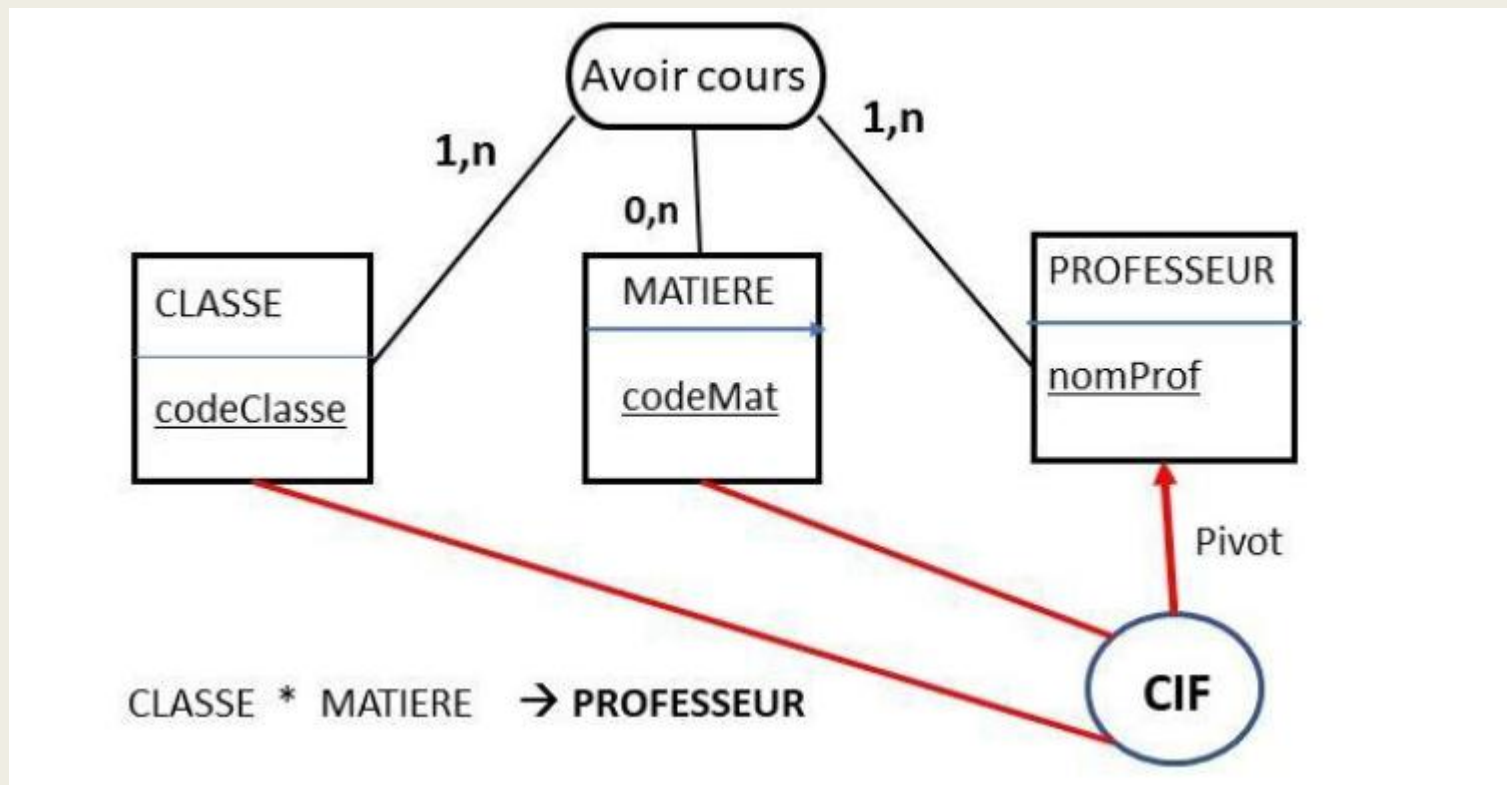
# Modèle Entités/Associations

Exemple : une matière n'est enseignée que par un seul professeur pour une classe donnée



# Modèle Entités/Associations

Exemple : une matière n'est enseignée que par un seul professeur pour une classe donnée



# Modèle Entités/Associations

- Relation réflexive : elle intervient lorsqu'une même entité est impliquée plusieurs fois dans une relation.
- Une relation binaire réflexive exprime par exemple :
  - un Contrat est l'avenant d'un Contrat précédent
  - un Projet est la suite d'un autre Projet
  - etc.

