

# **Introduction: La méthode MERISE**

Auteur: Jean-Baptiste Lavis

# **C'est quoi MERISE?**

**La méthode de conception d'un système  
d'information la plus utilisée en France**

# Pourquoi utiliser MERISE?

- **améliore la communication** entre tous les acteurs du processus de développement du projet
- **encadre le projet** grâce à des modèles
- **protège d'un développement hors sujet** potentiel

# Les outils

Dessin collaboratif:

[Excalidraw](#)

Diagrammes en ligne:

[Draw.io](#)

# Dépendances fonctionnelles

Données décrites dans un format générique:

- alphabétique (caractères)
- alphanumérique (des caractères, chiffres...)
- numérique (les nombres)
- date
- logique (0 ou 1, vrai-faux, oui ou non)

# Dictionnaire des données

Document qui recense toutes les dépendances fonctionnelles

Nom de la donnée	Format	Longueur
Prénom	alphabétique	30
Code postal	alphanumérique	10
Date d'embauche	date	

# Modèle Conceptuel de Données (MCD)

Décrit les données utilisées par le système  
d'information

Permet à un non informaticien de participer à l'écriture

Basé sur des **propriétés, entités et relations**

# Une propriété

## Informations élémentaires du SI

Exemple:

Un apprenant peut avoir un *nom*, un *prénom*, une *adresse*, etc...

NB: Disposera d'un type lors de la phase du MPD



# Une entité

Représentation d'un ensemble à conceptualiser

Regroupe plusieurs propriétés

**Doit posséder un identifiant**

apprenant
<u>id_apprenant</u>
nom
prénom
adresse

# Un mot sur l'identifiant

Nommé aussi **clé**

Propriété définie pour attribuer une valeur unique à  
chaque donnée

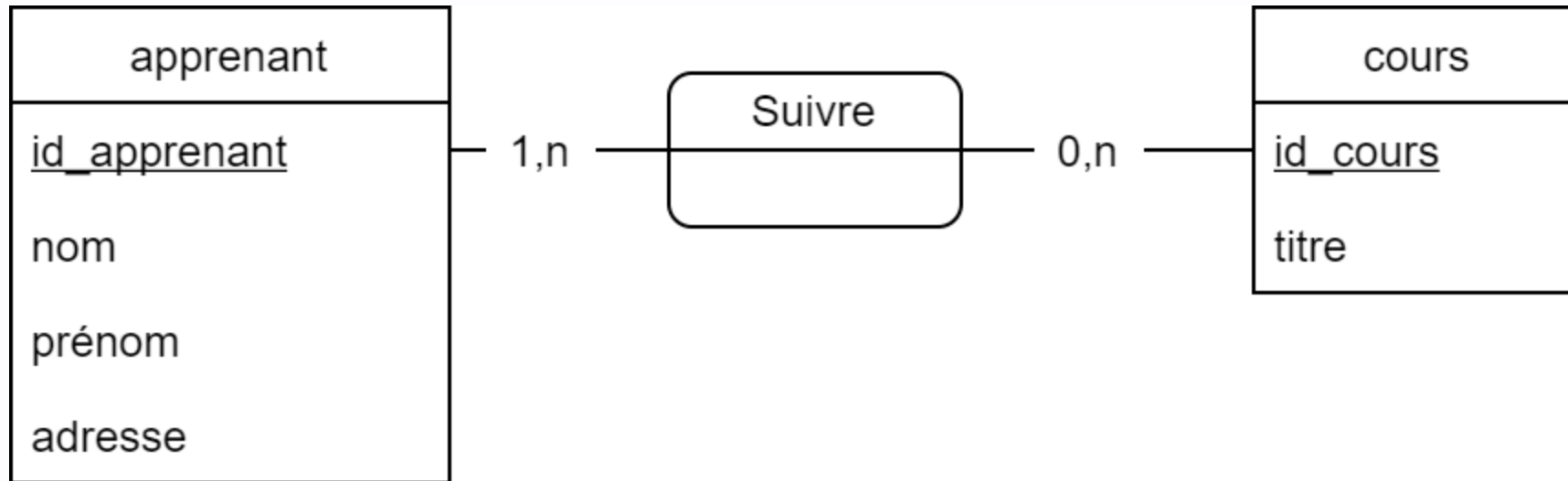
NB: on le souligne dans l'entité du MCD

# Les relations

Liaison des entités entre elles

Exemple:

Un *apprenant* peut suivre un *cours*



# Les Cardinalités

Exprime le nombre minimum et maximum de fois où  
l'entité participe à la relation

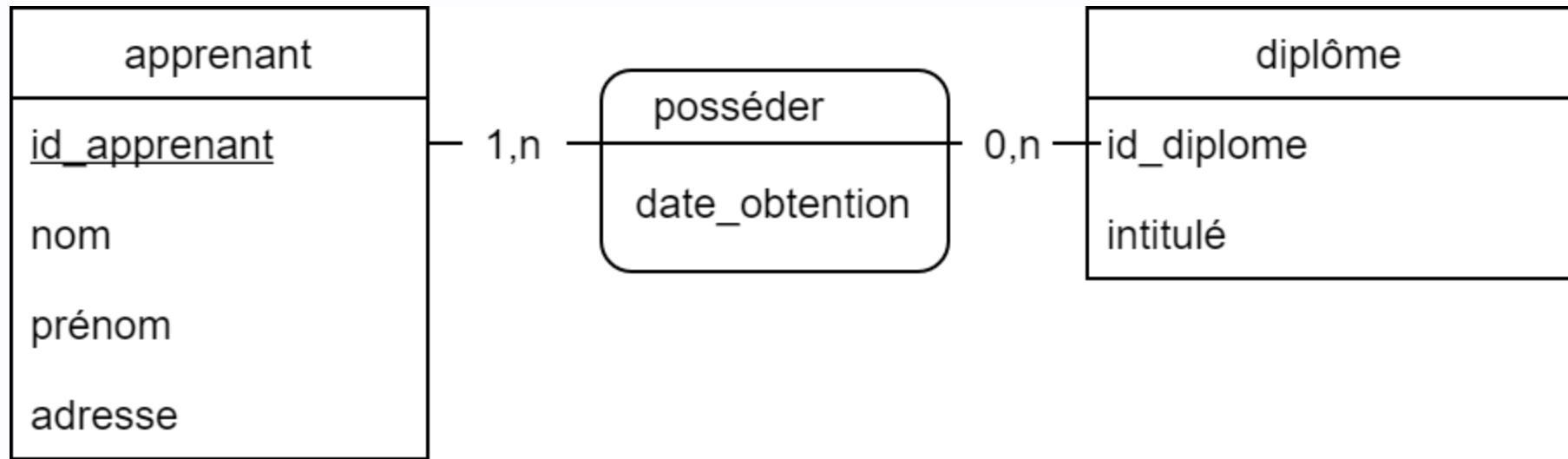
Exemple:

**Minimale**, l'apprenant peut suivre **1** cours

**Maximale**, l'apprenant peut suivre **n** (*plusieurs*) cours

# Les relations porteuses

Une relation est dite porteuse lorsqu'elle contient des propriétés.



# Modèle Logique de Données (MLD)

Découle directement du MCD

Les associations sont **transformés selon le type des relations:**

- entité faible vs entité forte
- entité forte vs entité forte
- entité faible vs entité faible

# Notions d'entité faible / forte

**Une entité est faible** si sa cardinalité maximale dans une relation est de 0 ou 1

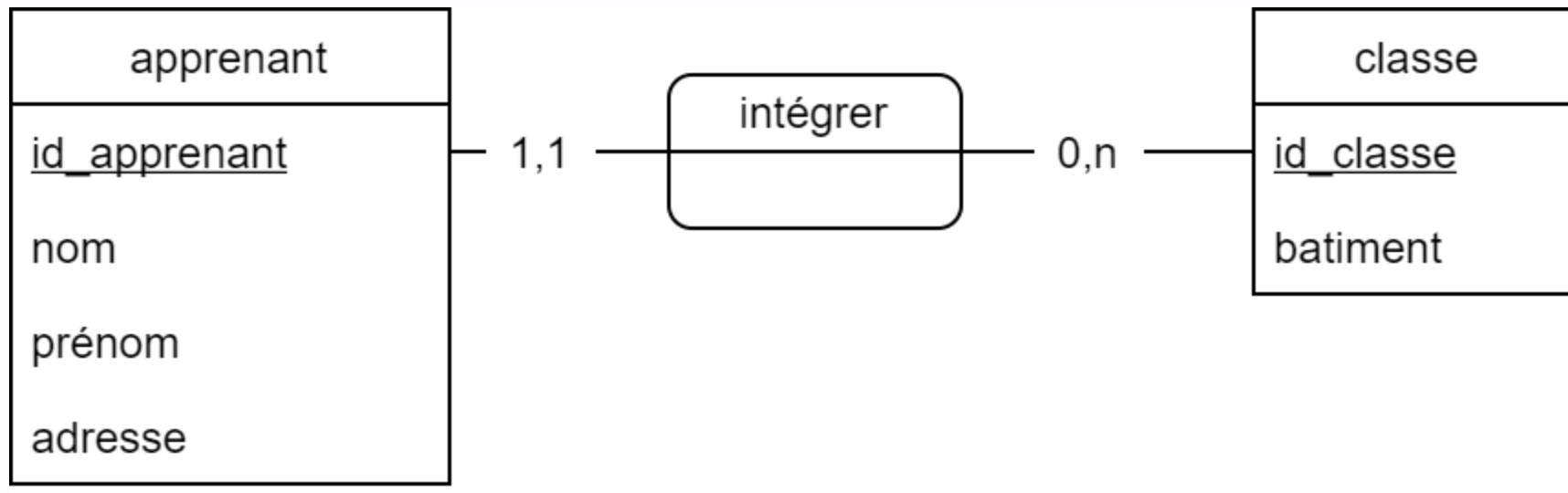
**Une entité est forte** si sa cardinalité maximale est de  $n$ .

# Entité faible vs entité forte

L'entité faible crée une liaison en copiant l'identifiant de l'entité forte (-> clé étrangère)

NB: Les clés étrangères sont représentés par un #





**devient**



# Entité forte vs entité forte

Une nouvelle entité apparaît à la place de la relation avec l'identifiant des deux entités fortes

apprenant
<u>id_apprenant</u>
nom
prénom
adresse

apprenant_diplome
# id_apprenant
# id_diplome
date_obtention

diplôme
<u>id_diplome</u>
intitulé

# Entité faible vs entité faible

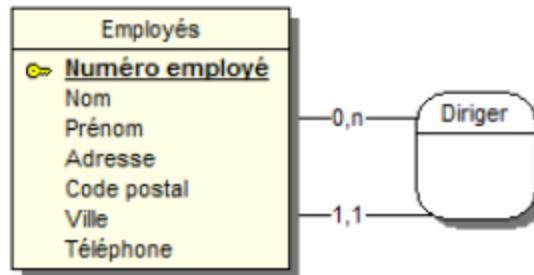
Deux possibilités:

- Fusionner les propriétés des deux tables dans une seule
- Si on veut conserver deux tables, choix de laquelle portera une clé étrangère

# La relation réflexive

Cas plus rare :

**Quand une entité est associée à elle-même.**



Dans ce cas, on crée une clé étrangère dans la même table (ex: id\_manager)

# Modèle Physique de Données (MPD)

- Les entités vont devenir des tables
- Toutes les propriétés vont obtenir un type de données (varchar, int, date, etc...)
- Les identifiants vont devenir des clé primaires ou étrangères

apprenant
<u>PK id_apprenant</u> int
nom varchar(30)
prénom varchar(30)
adresse varchar(100)

apprenant_diplome
FK id_apprenant int
FK id_diplome int
date_obtention date

diplôme
<u>PK id_diplome</u> int
intitulé varchar(40)