

Base de Données

Les bases de données sont omniprésentes :

- Hautes Écoles, Universités :
 - *Données sur les étudiants*
 - *Id, nom, prénom, classe, section, cycle, année...*
 - *Données sur les formations*
 - *matière, intervenant (enseignant), public, salle*
 - *Données sur les résultats*
 - *matière, intervenant (enseignant), pondération, résultat*
- Entreprises
 - *fichiers clients, fournisseurs, commandes*
 - *facturation,*
 - *gestion de stock, inventaire.*

Pourquoi pas les fichiers

L'utilisation de fichiers impose à l'utilisateur de connaître :
le mode d'accès (séquentielle, indexée, ...)
la structure physique des enregistrements et leur localisation .

Ecriture de nouveaux programmes pour des nouvelles applications, (création de nouveaux fichiers qui contiendront peut-être des informations déjà présentes dans d'autres fichiers).

Réécriture des programmes qui manipulent les fichiers après modification de la structure d'un enregistrements (ajout d'un champ par exemple).

Pourquoi pas les fichiers

- applications
 - rigides,
 - contraignantes
 - longues et coûteuse à mettre en œuvre
- Les données associées sont :
 - mal définies et mal désignées,
 - redondantes
 - peu accessibles de manière ponctuelle
 - peu fiables

Notions de bases

Définition intuitive d'une BD (1)

- Définition intuitive : on peut considérer une **Base de Données** (BD) comme une grande quantité de données (ou ensemble d'informations), centralisées ou non, servant pour les besoins d'une ou plusieurs applications, interrogeables et modifiables par un groupe d'utilisateurs travaillant en parallèle.

Notions de bases

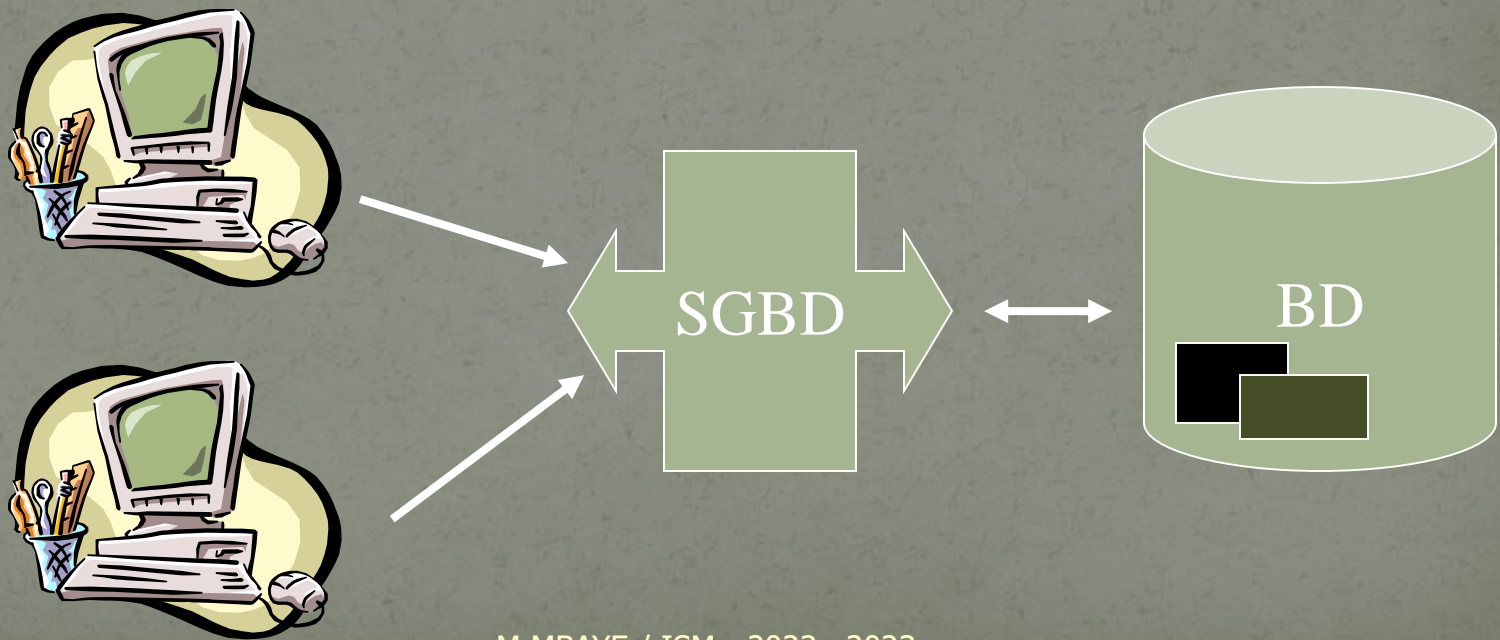
Définition intuitive d'une BD (2)

- **Définition** ; une base de données est un ensemble structuré de données (1) enregistrées sur des supports accessibles par l'ordinateur (2) pour satisfaire simultanément plusieurs utilisateurs (3) de manière sélective (4) en un temps opportun (5).

- (1) : Organisation et description de données
- (2) : Stockage sur disque
- (3) : Partage des données
- (4) : Confidentialité
- (5) : Performance

SGBD (1)

Définition : Le logiciel qui permet d'interagir avec une BD est Système de Gestion de Base de Données (SGBD)



Objectifs des SGBD

- **Indépendance physique** : La manipulation des données doit être facilitée en travaillant directement sur le schéma logique. On peut insérer, supprimer, modifier des données directement sur l'image logique. Le SGBD va s'occuper de faire le travail sur les fichiers physiques.
- **Indépendance logique** : Un même ensemble de données peut être vu différemment par des utilisateurs différents. Toutes ces visions personnelles des données doivent être intégrées dans une vision globale.
- **Manipulations des données par des non informaticiens**. Il faut pouvoir accéder aux données sans savoir programmer ce qui signifie des langages « quasi naturels ».
- **Efficacité des accès aux données** : Ces langages doivent permettre d'obtenir des réponses aux interrogations en un temps « raisonnable ». Il doivent donc être optimisés et, entre autres, il faut un mécanisme permettant de minimiser le nombre d'accès disques. Tout ceci, bien sur, de façon complètement transparente pour l'utilisateur.

Objectifs des SGBD

- Administration centralisée des données : Des visions différentes des données (entre autres) se résolvent plus facilement si les données sont administrées de façon centralisée.
- Cohérence des données. Les données sont soumises à un certain nombre de contrainte d'intégrité qui définissent un état cohérent de la base. Elles doivent pouvoir être exprimées simplement et vérifiées automatiquement à chaque insertion, modification ou suppression de données, par exemple :
 - l'âge d'une personne supérieur à zéro
 - Salaire supérieur à zéro
 - Etc

Dés que l'on essaie de saisir une valeur qui ne respecte pas cette contrainte, le SGBD le refuse.

Objectifs des SGBD

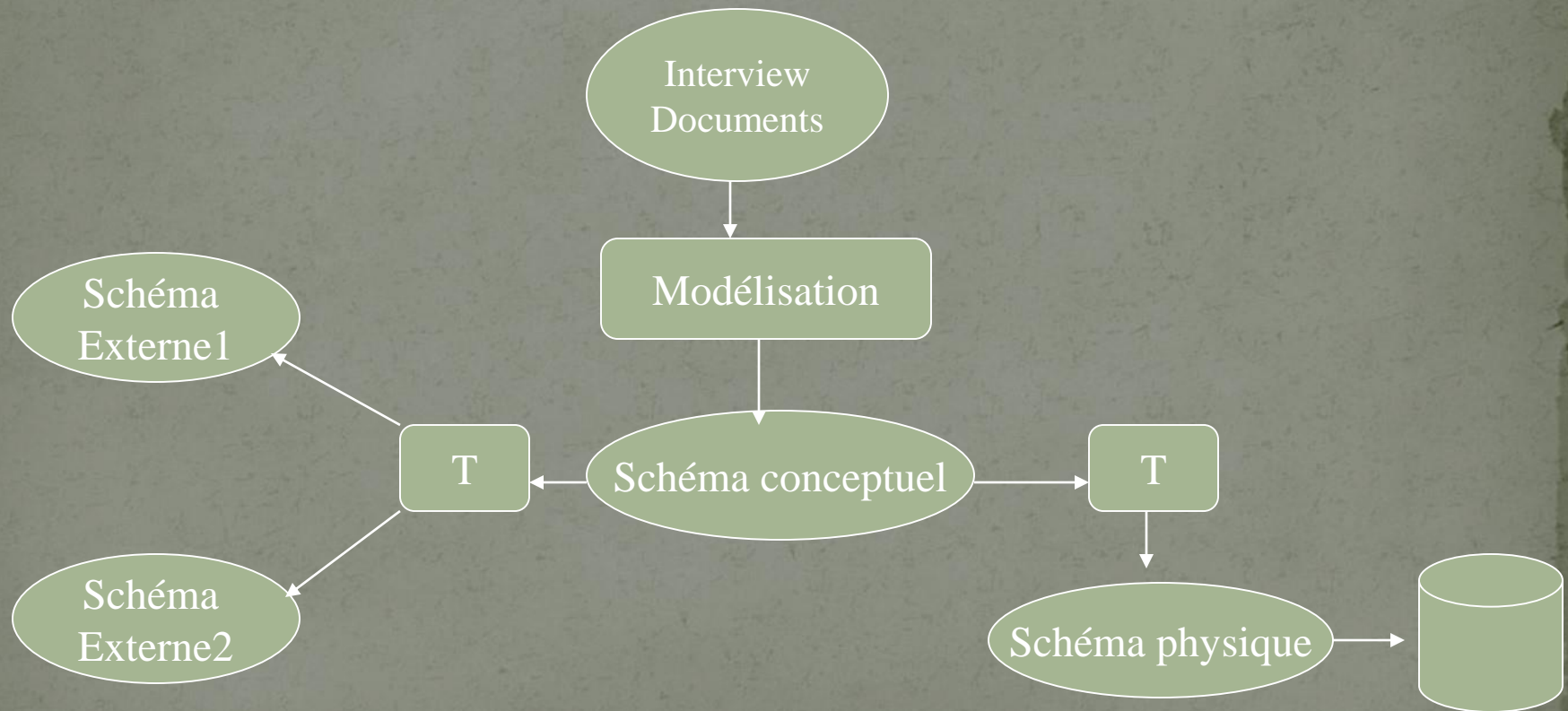
- **Non redondance des données** : Afin d'éviter les problèmes lors des mises à jour, chaque donnée ne doit être présente qu'une seule fois dans la base.
- **Partageabilité des données** : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux mêmes données au même moment. Si ce problème est simple à résoudre quand il s'agit uniquement d'interrogations et quand on est dans un contexte mono-utilisateur, cela n'est plus le cas quand il s'agit de modifications dans un contexte multi-utilisateurs. Il s'agit alors de pouvoir :
 - Permettre à deux (ou plus) utilisateurs de modifier la même donnée « em même temps »;
 - Assurer un résultat d'interrogation cohérent pour un utilisateur consultant une table pendant qu'un autre la modifie.

Objectifs des SGBD

- **Sécurité des données**. Les données doivent pouvoir être protégées contre les accès non autorisés. Pour cela, il faut pouvoir associer à chaque utilisateur des droits d'accès aux données.
- **Résistance aux pannes** : Que se passe-t-il si une panne survient au milieu d'une modification, si certains fichiers contenant les données deviennent illisibles? Les pannes, bien qu'étant assez rares, se produisent quand même de temps en temps. Il faut pouvoir, lorsque l'une d'elles arrive, récupérer une base dans un état « sain ». Ainsi, après une panne intervenant au milieu d'une modification deux solutions sont possibles : soit récupérer les données dans l'état dans lequel elles étaient avant la modification, soit terminer l'opération interrompue.

Définition et description des données

3 niveaux de description



Types d'utilisateurs d'une base de données

- **L'administrateur** de la base est chargé du contrôle de la base de données. Il est chargé de permettre l'accès aux données aux applications ou individus qui y ont droit et de conserver de bonnes performances d'accès à ces données. Il est aussi chargé des sauvegardes et des procédures de reprise après les pannes.
- **Le programmeur d'applications** utilise la base de données pour construire ses applications. Il a droit de créer de nouvelles tables et les structures associées (vues, index, cluster, etc.). Il définit avec l'administrateur de la base les droits qui seront accordés aux utilisateurs des applications qu'il développe.

L'utilisateur final n'a accès qu'aux données qui lui sont utiles. L'administrateur de la base peut lui accorder certains droits : consultations, modification, suppression des données.

Modèles de SGBD

- Quelques modèles logiques :

- **Modèle hiérarchique**

Les données sont représentées sous forme d'une structure arborescente d'enregistrements.

Cette structure est conçue avec des pointeurs et détermine le chemin d'accès aux données.

- **Modèle réseau**

La structure des données peut être visualisée sous la forme d'un graphe quelconque.

Comme pour le modèle hiérarchique, la structure est conçue avec des pointeurs et détermine Le chemin d'accès aux données.

Pour ces deux modèles, les programmes sont dépendants de la structure logique de la base et du chemin d'accès aux données : ils doivent décrire comment retrouver les données (on parle de navigation dans la base) et si, par exemple, on enlève un index, tous les programmes qui l'utilisaient doivent être réécrits. De plus le langage de travail est complexe.

Modèles de SGBD

- **Modèle relationnel**

Il est fondé sur la théorie mathématique des relations.

- **Modèle objet**

Les données sont représentées sous forme d'objets au sens donné par les langages orientés objet : elles sont enregistrées avec les procédures et fonctions qui permettent de les manipuler.

- **Quelques SGBD (relationnels du marché)**

- **Micro** : ACCESS, PostgreSQL, MySQL, ...
- **Gros système** : DB2, ORACLE, SYBASE, ...

Modèle Entité-Association

L'objectif du modèle Entité-Association est de permettre de représenter graphiquement les ensembles de données et leurs liens sémantiques sans se préoccuper du Modèle organisationnel (ou logique), des données (hiérarchique, réseau, relationnel, ...) et encore moins du SGBD choisi pour implémenter la BD (Modèle Physique). La finalité de ce modèle est d'obtenir le modèle conceptuel des données (MCD).

Modèle Entité-Association(MCD)

Le but du MCD est de modéliser (formaliser) les données mémorisées du S.I On ne tient pas compte des aspects techniques, économiques, ni des problèmes du stockage de l'information, ni des problèmes d'accès aux informations et ni des conditions d'utilisation.

DEMARCHES

I. TECHNIQUE DE CHEN

Cette démarche est aussi appelé modélisation directe.

Les individus et les relations sont repérés directement à partir du discours ; un nom devient un individu et un verbe une relation.

DEMARCHES

II. MODELISATION ASCENDANTE

Dans cette démarche on établit **le Dictionnaire de Données** à partir des **documents** disponibles.

On parle souvent de **règles de gestions**

DEMARCHES

II. 1 / Epuration du dictionnaire de données

- Synonymes :

noms différents recouvrant la même propriété
(salarié et employé)

- Polysémies :

Même nom pour deux informations différentes
(date pour date-facture et date-commande)

DEMARCHES

II. 2 / Les dépendances fonctionnelles(DF)

Il y a dépendance fonctionnelle entre deux propriétés lorsque la connaissance d'une valeur d'une propriété permet de déterminer une et une seule valeur d'une autre propriété ;

La dépendance fonctionnelle est notée $P_1 \rightarrow P_2$

On lit P_1 détermine P_2 ou P_2 dépend P_1

DEMARCHES

II. 3 / Graphe des dépendances fonctionnelles (GDF)

- On établit la liste des DF
- On ordonne la liste des DF, pour avoir une vision synthétique

DEMARCHES

II. 4 / Modèle conceptuel des données brut

- A partir du GDF on établit le MCD brut

Modèle Entité-Association(MCD)

I. Le concept d'entité

Une entité est une population d'individu homogènes.

Par exemple les matières d'une même filière peuvent être représenté par une même entité.

D'une matière à l'autre, les informations ne changent pas la nature de la matière seul leurs valeurs peuvent changer.

Modèle Entité-Association(MCD)

➤ Formalisme

Nom de l'entité

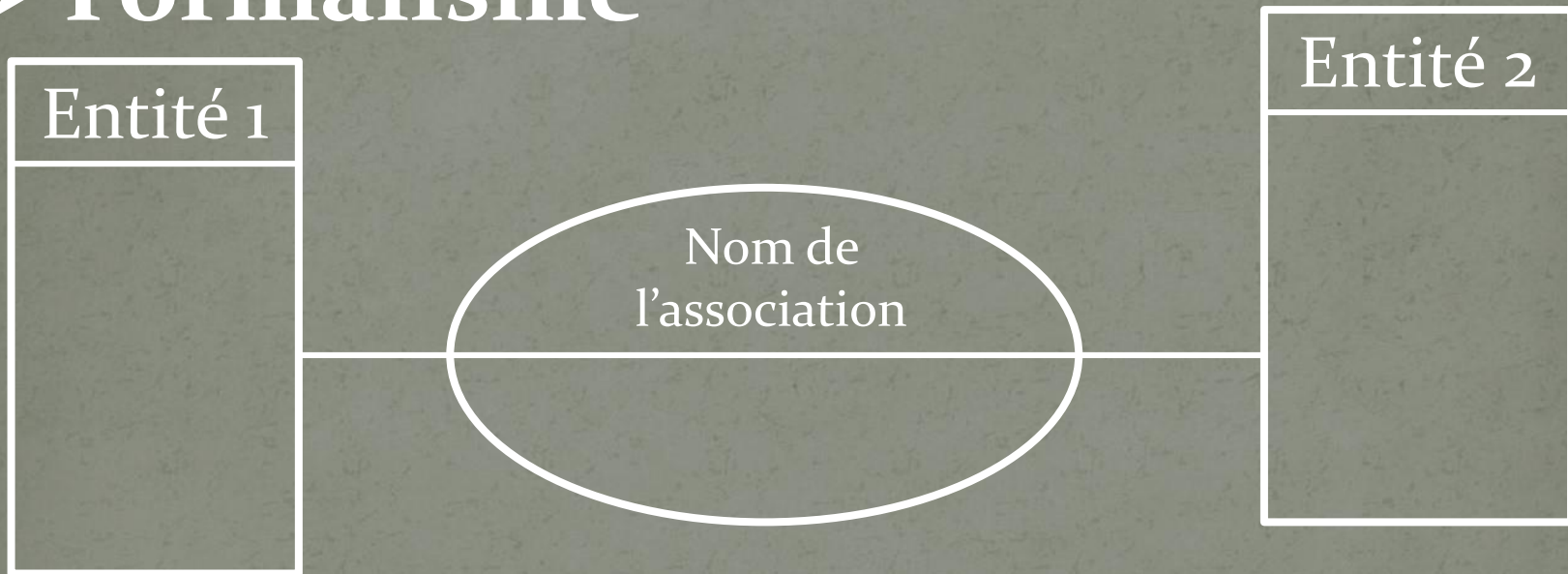
Modèle Entité-Association(MCD)

II. Le concept d'association

Une association est une liaison ayant une signification bien précise entre un ou plusieurs entités.

Modèle Entité-Association(MCD)

➤ Formalisme



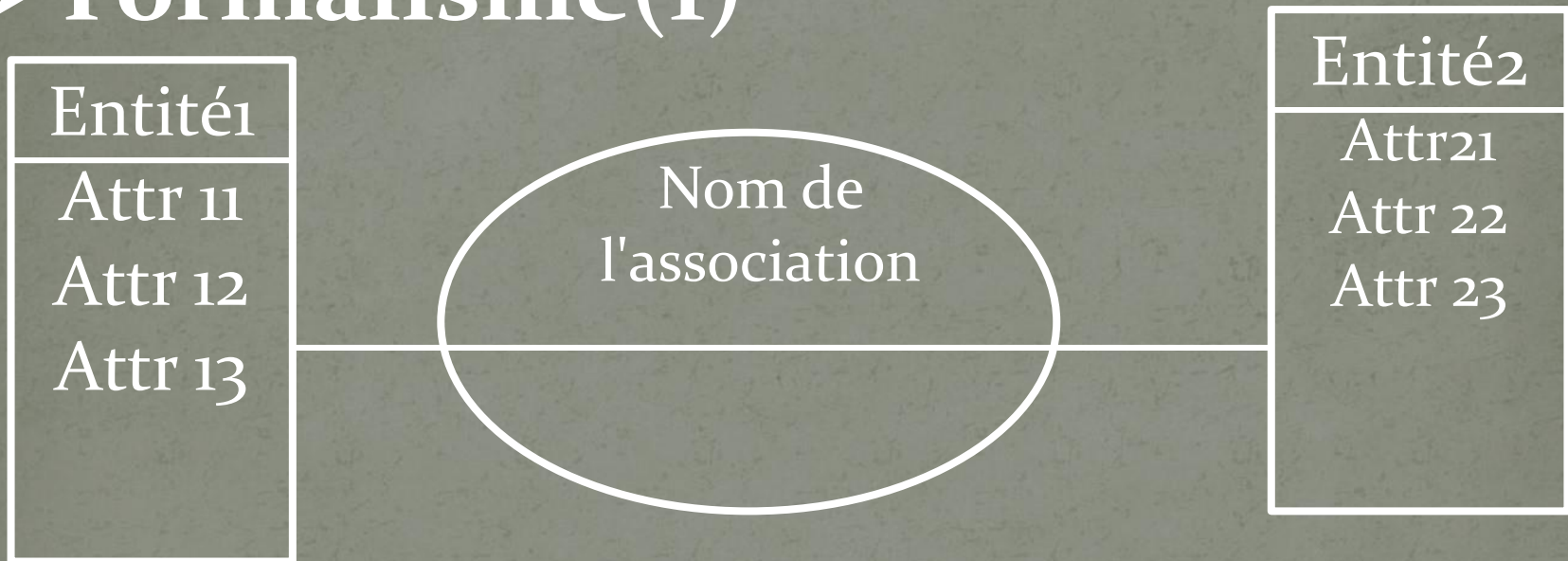
Modèle Entité-Association(MCD)

III.Le concept d'attribut

Une attribut est une propriété d'une entité ou d'une association.

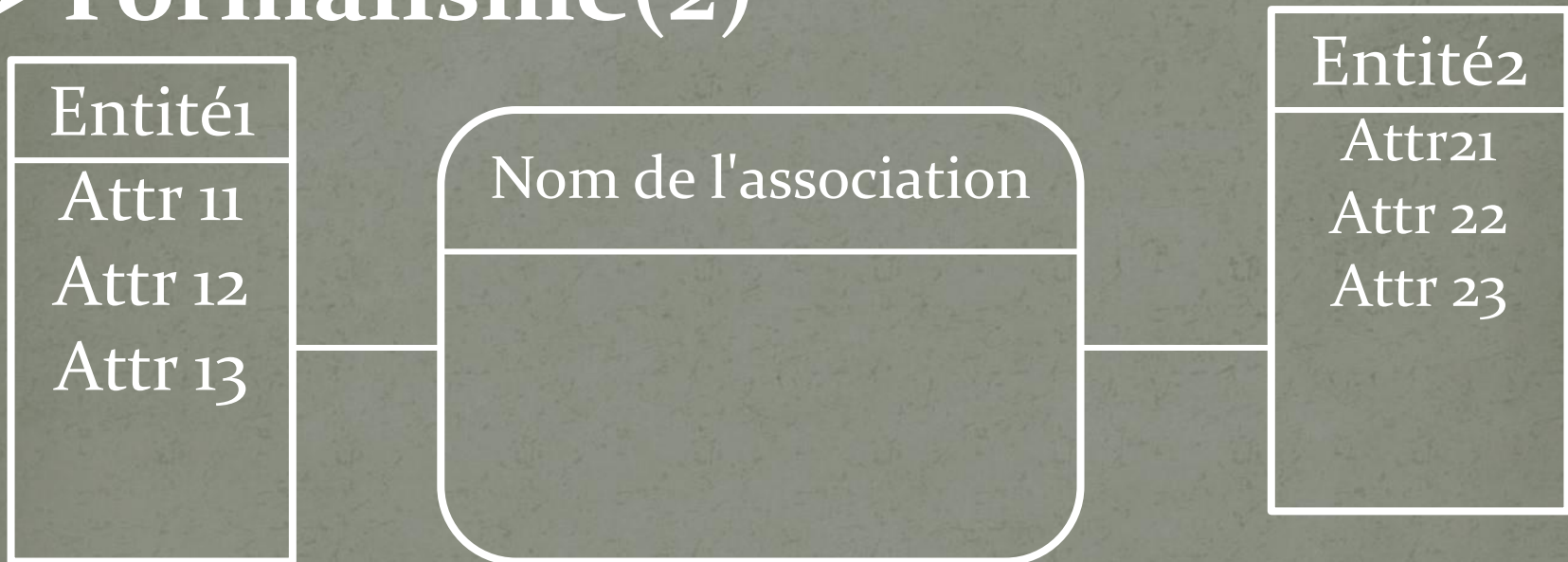
Modèle Entité-Association(MCD)

➤ Formalisme(1)



Modèle Entité-Association(MCD)

➤ Formalisme(2)



Modèle Entité-Association(MCD)

III. Le concept d'identifiant

Un identifiant est une propriété ou groupe de propriétés qui sert à identifier une entité.

Elle est choisie par l'analyste de façon à ce que deux occurrences de cette entité ne puissent pas avoir le même identifiant.

- Une entité possède au moins un attribut (son identifiant).
- Une association peut être dépourvue d'attribut.

Modèle Entité-Association(MCD)

➤ Formalisme

Entité
<u>Identifiant</u>
Attr 11
Attr 12
Attr 13

Modèle Entité-Association(MCD)

Le concept de cardinalité

La cardinalité d'une entité par rapport à une association s'exprime par deux nombres: la cardinalité minimale et cardinalité maximale. Elles expriment le nombre de valeurs minimum et maximum que l'entité peut prendre dans l'association.

Modèle Entité-Association(MCD)

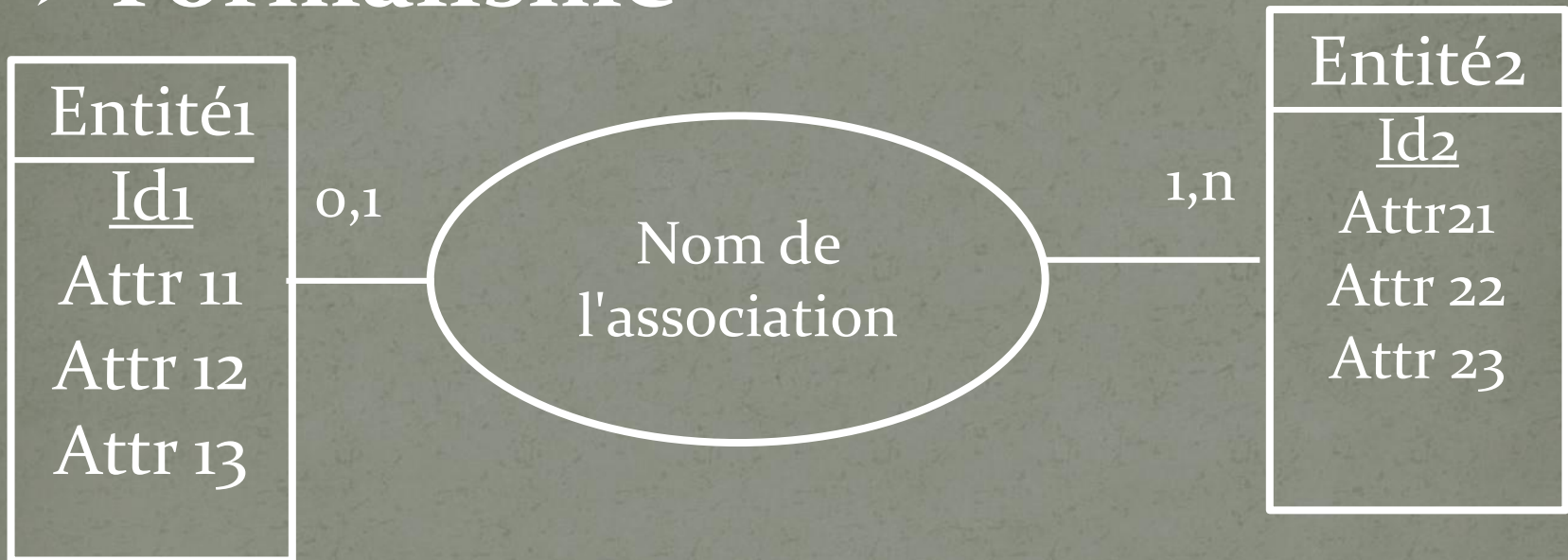
La cardinalité minimale est dénombrable, c'est le nombre de fois minimum qu'une occurrence d'une entité participe aux occurrences de la relation. La cardinalité minimale est égale à 1 ou 0.

La cardinalité maximale est le nombre de fois maximum qu'une occurrence d'une entité participe aux occurrences de la relation La cardinalité maximale est égale à 1 ou tout nombre fixé (n).

En d'autres termes, les cardinalités désignent le minimum et le maximum de fois qu'une occurrence de l'entité peut être concernée par l'association.

Modèle Entité-Association(MCD)

➤ Formalisme



Modèle Entité-Association(MCD)

Cardinalité : traduction

0,1 : Aucun ou un

1,1 : Un et un seul

0,n : Aucun ou au moins un

1,n : Au moins un

Règles de normalisation du MCD

La normalisation permet :

- ✓ De limiter les redondances de données,
- ✓ De limiter les pertes de données,
- ✓ De limiter les incohérences au sein des données et
- ✓ D' améliorer les performances des traitements.

Un bon schéma entités-associations doit répondre à neuf (9) règles de normalisation.

Règles de normalisation du MCD

✓ Normalisation des entités :

Toutes les entités qui sont remplaçables par une association doivent être remplacées (Exemple d'association multiple).

✓ Normalisation des noms :

le nom d'une entité, d'une association ou d'un attribut doit être unique.

Des entités homogènes peuvent être fusionnées.

Exemple (enseignant, étudiant, directeur ...)

Règles de normalisation du MCD

✓ Normalisation des identifiants :

chaque entité doit posséder un identifiant.

L'identifiant doit être un entier de préférence incrémenté automatiquement.

✓ Normalisation des attributs :

Remplacer les attributs en plusieurs exemplaires en une association supplémentaire de cardinalités maximales n et ne pas ajouter d'attribut calculable à partir d'autres attributs.

Exemple (si plusieurs Numéro créer l'entité numéro)

les attributs d'une association doivent dépendre directement des identifiants de toutes les entités en association.

Règles de normalisation du MCD

✓ Normalisation des associations:

Il faut éliminer les associations fantômes, redondantes ou en plusieurs exemplaires.

✓ Normalisation des cardinalités :

Une cardinalité minimale est toujours 0 ou 1.

Une cardinalité maximale est toujours 1 ou n .

Les formes normales

A ces 6 règles de normalisation, il convient d'ajouter les 3 premières formes normales.

Les formes normales correspondent à une décomposition optimale des entités en relations élémentaires correctement construites. Cela évite les redondances d'informations et facilite la maintenance des données.

Les formes normales

➤ Première forme normale

Une relation est en première forme normale si tout attribut contient une valeur atomique.

C'est à dire (non multiples, non composées)
Chaque entité doit posséder un identifiant (une clé) qui la caractérise de manière unique.

Les formes normales

➤ Deuxième forme normale :

Une relation est en deuxième forme normale si et seulement si :

- elle est en première forme normale ;
- tout attribut n'appartenant pas à une clé ne dépend pas que d'une partie de cette clé.

L'identifiant peut être composé de plusieurs attributs mais les autres attributs de l'entité doivent dépendre de l'identifiant en entier (et non pas une partie de cet identifiant).

Les formes normales

➤ Troisième forme normale de Boyce-Codd :

Une relation est en troisième forme normale si et seulement si :

- elle est en deuxième forme normale ;
- tout attribut n'appartenant pas à une clé ne dépend pas d'un attribut non-clé.

Tous les attributs d'une entité doivent dépendre directement de son identifiant et d'aucun autre attribut.