

Systèmes d'Information et Bases de Données Cycle Ingénieur 1^{ère} Année

Abdelhak BOULAALAM

<http://boulaalam.usmba.ac.ma>

abdelhak.boulaalam@usmba.ac.ma

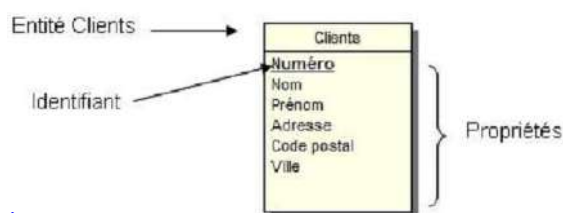
Semaine 3

Mersie – conception SIO -

MCD

■ Entité - L'identifiant ou key (clé)

NomEntité
identifiant
nom_attribut 1
nom_attribut 2
etc.

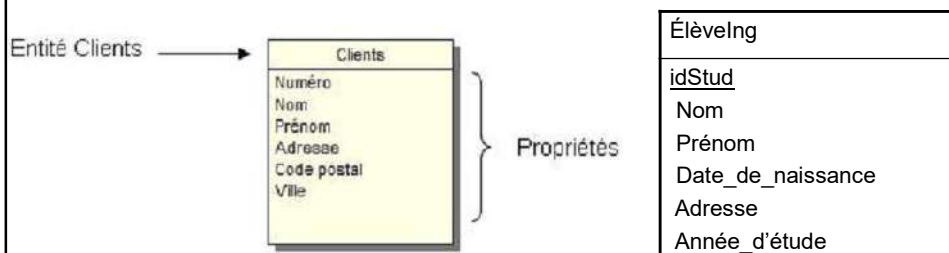


■ Règles - un identifiant doit être :

- ☐ **Univalué et non nul** : à une occurrence de l'entité correspond **une et une seule valeur** d'un identifiant;
- ☐ **Discriminant** : à une valeur d'un identifiant correspond **une et une seule occurrence** de l'entité ;
- ☐ **Stable** : une occurrence donnée d'entité doit conserver la même valeur d'identifiant jusqu'à sa destruction ;
- ☐ **Minimal** : si l'identifiant est élémentaire il est nécessairement minimal. Dans le cas d'un identifiant **composé**, pour que celui-ci soit minimal, il faut que la suppression d'un de ses composants lui fasse perdre son caractère **discriminant**.

■ Entité (Entity)

- L'entité et ses propriétés sont schématisés graphiquement comme suit (pour la méthode Merise).
 - Note: Il existe d'autres schématisations:
 - Par exemple le model **Entity-Relationship Model**



■ Le Concept : Entité, Ensembles d'entités

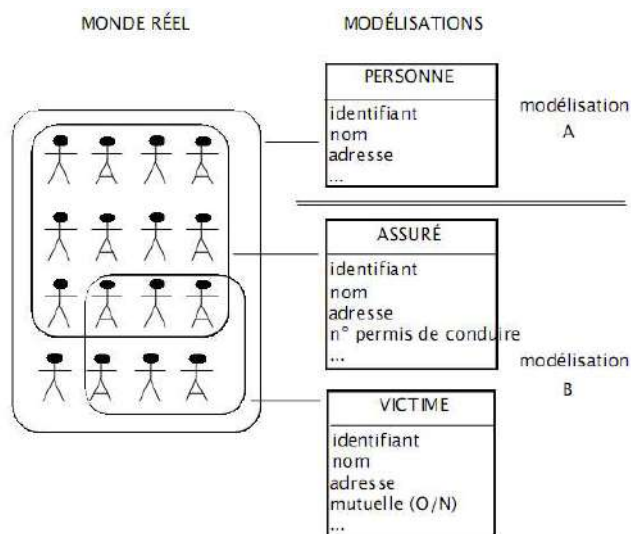
- Entité/Entity: est une « **chose** » ou un « **objet** » dans le monde réel qui se distingue de tous les autres objets.
 - Par exemple:
 - **chaque personne** dans une université est une **entité**.
 - Chaque **cours** dans l'ENSA est une **entité**.
 - Une entité:
 - Possède un ensemble de **propriétés**.
 - Les valeurs d'un ensemble de propriétés doivent **identifier de manière unique une entité**.
 - Par exemple, une personne peut avoir une propriété identifiant idP
 - Ainsi, la valeur 677-89-9011 de idP identifierait de manière unique une personne en particulier dans l'université.
 - Une entité peut être **concrète**, comme une personne ou un livre, ou **abstraite**, comme un cours, une offre de cours ou une réservation de vol.

■ Concept : Entité, Ensembles d'entités

- Un ensemble d'entités est un ensemble d'entités du même type qui **partagent les mêmes propriétés ou attributs**.
 - Exemple 1: L'ensemble de toutes les personnes qui sont instructeurs dans une université peut être défini comme l'ensemble d'entités **instructeur**.
 - Exemple 2: De même, l'ensemble d'entités **étudiant** peut représenter l'ensemble de tous les étudiants.
- Les ensembles d'entités n'ont pas besoin d'être **disjoints (XOR)**
 - Par exemple, il est possible de définir l'ensemble d'entités personne composé de toutes les personnes d'une université.
 - Une entité personne peut être une entité instructeur, une entité étudiant, les deux ou aucune.

■ Entité - objet - Individu

- **Différentes modélisations possibles.**



■ Concept : Entité, Ensembles d'entités (checkpoint)

- ☐ Une entité est représentée par un **ensemble d'attributs**.
- ☐ Les attributs sont des **propriétés descriptives** possédées par **chaque membre** d'un ensemble d'entités.
- ☐ Un attribut appartient à un domaine
 - Domaine : un ensemble de valeurs
- ☐ Chaque entité peut avoir sa propre valeur pour chaque attribut.
- ☐ Exemple:
 - Les attributs possibles de l'ensemble d'entités d'instructeur/professeur sont l'**ID**, le nom, une spécialité, le nom du département et le salaire ... etc. (Cahier de charge DD Projet de fin de module)

■ Design Process

- ☐ Créer une application de base de données implique:
 1. La conception du schéma de la base de données,
 2. La conception des programmes qui accèdent et mettent à jour les données,
 3. La conception d'un schéma de sécurité pour contrôler l'accès aux données.
- ☐ Les besoins des utilisateurs jouent un rôle central dans le processus de conception.
 - Phase analyse ... Conception : DDC

■ Design Process ... En 3 phases

□ Phase initiale -

- caractériser pleinement les besoins en données des utilisateurs potentiels de la base de données.

□ Deuxième phase - choisir un modèle de données

- Application des concepts du modèle de données choisi
- Traduire ces exigences en un schéma conceptuel de la base de données.
- Un schéma conceptuel pleinement développé indique les besoins fonctionnels de l'entreprise.
 - Décrivez les types d'opérations (ou transactions) qui seront effectués sur les données.

■ Design Process ... En 3 phases

□ Phase finale - Passer d'un modèle de données abstrait à la mise en œuvre de la base de données

- Conception logique - Décider du schéma de la base de données.
 - La conception de la base de données exige que nous trouvions une "bonne" collection de schémas de relations.
 - Décision commerciale - Quels attributs devons-nous enregistrer dans la base de données?
 - Décision en informatique - Quels schémas de relations devons-nous avoir et comment les attributs doivent-ils être répartis parmi les différents schémas de relations?
- Conception physique - Décider de la disposition physique de la base de données.

■ Design Process

□ Alternatives de conception (Design Alternatives)

- En concevant un schéma de base de données, nous devons nous assurer d'éviter deux écueils majeurs :
 - **Redondance** : une mauvaise conception peut entraîner la répétition d'informations.
 - La représentation redondante des informations peut entraîner une incohérence des données entre les différentes copies d'informations.
 - **Incomplétude** : une mauvaise conception peut rendre certains aspects de l'entreprise difficiles ou impossibles à modéliser.
- Éviter les mauvaises conceptions ne suffit pas. Il peut y avoir un grand nombre de bonnes conceptions parmi lesquelles nous devons choisir.

■ Concept : Entité, Ensembles d'entités ([checkpoint](#))

Exercice TD

- [Exercice d'application](#): Faire une description des entités Etudiant, Professeur et Cours de l'ENSA.

Spécifications pour le cas d'étude (Projet de fin de module)

Les enseignants de l'USMBA sont identifiés par un ID, un nom, un prénom, le nom du département d'affectation et un salaire. En outre, d'autres attributs et entités sont à prendre en compte.

Un module peut être composé de plusieurs éléments de module. Chaque module possède un identifiant, un titre, le nom de la filière, le nom du département, ainsi qu'un statut indiquant s'il s'agit d'un module de base ou d'un module supplémentaire.

TAR1 : Contacter le service de scolarité afin d'obtenir le cahier des charges et les règles de gestion concernant les enseignants et les cours.

■ Concept : Entité, Ensembles d'entités

□ Chaque entité a une valeur pour chacun de ses attributs... une instance

■ Par exemple, une entité d'instructeur particulière peut avoir la valeur:

- 12121 pour l'ID,
- la valeur Boulaalam pour le nom,
- la valeur Informatique pour le département
- et la valeur 10000 pour le salaire
-

■ Concept : Entité, Ensembles d'entités

- L'attribut **ID** est utilisé pour identifier les instructeurs de manière unique.
- Remarque:
 - Parfois il faut utiliser un numéro d'identification délivré par le gouvernement comme un attribut dont la valeur identifie de manière unique la personne.
 - Cependant, cela est considéré comme une mauvaise pratique pour des raisons de sécurité et de confidentialité.
 - Aussi la possibilité d'inventer un identificateur (cas échéant)

■ Concept : Entité, Ensembles d'entités

- Le model conceptuel ... Schématisation
 - Diagramme Merise MCD E-A(Entité-Association) /E-R montrant les deux ensembles d'entités

Ensembles d'entités --
enseignant et étudiant

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

instructor

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

Entités -- enseignant et étudiant

instructor
<u>ID</u>
Name
salary

student
<u>ID</u>
Name
...

■ Entité - **L'identifiant**

- l'identifiant nommé aussi la clé → Il a un rôle bien précis:
 - L'identifiant permet de connaître de **façon sûre et unique** l'ensemble des propriétés qui participent à l'entité. (voir la partie DF)
- Exemple
 - le fait de connaître la ville d'un client permet-il de connaître son nom?
 - La réponse est non.
 - La connaissance du nom du client permet-elle de connaître sa ville?
 - La réponse est toujours non, car en cas d'homonymie.
- Solution**
- Il faut donc **trouver**, ou **inventer**, une propriété qui lorsque sa valeur est connue permet la connaissance de l'ensemble des valeurs qui s'y rattachent de façon formelle.
 - Voir partie graphe de dépendances fonctionnelles
- Au niveau du formalisme, cette propriété se souligne.

- Entité - L'identifiant ou key (clé) ... Rappel 4 caractéristiques
- Règles - un identifiant doit être :
 - **Univalué et non nul** : à une occurrence de l'entité correspond **une et une seule valeur** d'un identifiant;
 - **Discriminant** : à une valeur d'un identifiant correspond **une et une seule occurrence** de l'entité ;
 - **Stable** : une occurrence donnée d'entité doit conserver la même valeur d'identifiant jusqu'à sa destruction ;
 - **Minimal** : si l'identifiant est élémentaire il est nécessairement minimal. Dans le cas d'un identifiant **composé**, pour que celui-ci soit minimal, il faut que la suppression d'un de ses composants lui fasse perdre son caractère **discriminant**.

Démarche à suivre:

➤ Etape 1:

- ✓ Elaborer un cahier de charge!
- ✓ Elaborer le DDC
- ✓ Discuter les DFs.
- ✓ Identifier les entités possible

➤ Etape 2:

- ✓ Schématiser l'ensemble des entités trouvées.

Mersie – conception SIO -

MCD – Associations / Relationship Sets

■ L'**association** (ou relation) / **Relationship**

- Merise MCD : c'est un modèle **Entité Association**
- E-R: Entity Relationship
- Une relation est une **association entre plusieurs entités**.

■ Example:

44553 (Peltier)	<i>advisor</i>	22222 (<u>Einstein</u>)
<i>student entity</i>	relationship set	<i>instructor entity</i>

- nous pouvons définir **association superviseur** qui associe l'instructeur X l'étudiant Y.
- Cette relation précise que X est un encadrant/superviseur de l'étudiant Y.
- Un **ensemble de relations**:
 - est un ensemble de relations du même type.
 - est une relation mathématique parmi $n \geq 2$ entités, chacune provenant d'ensembles d'entités (E)

$$\{(e_1, e_2, \dots, e_n) / e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

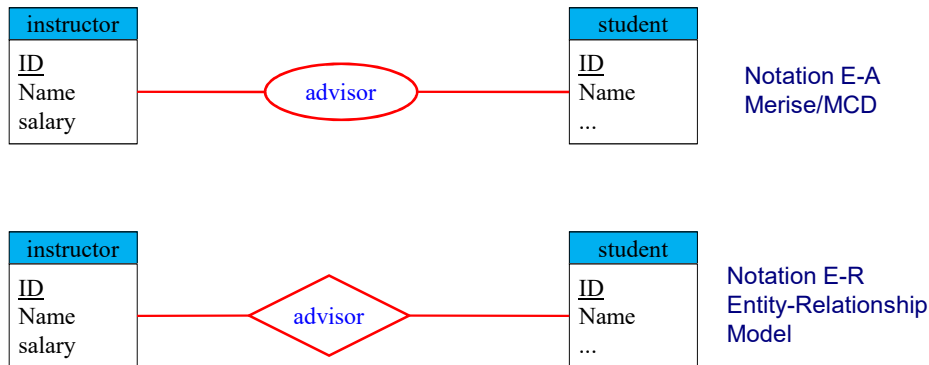
Où (e_1, e_2, \dots, e_n) est une relation

- Example:
 $(44553, 22222) \in \text{advisor}$

MCD - Relationship Sets

■ L'association (ou relation) / Relationship

- Une relation est une **association entre plusieurs entités**.
 - Nous définissons l'ensemble de relations supervise/advisor/Encadrer pour désigner les associations entre les étudiants et les instructeurs qui agissent comme leurs encadrant. ... **Schématisation ??**

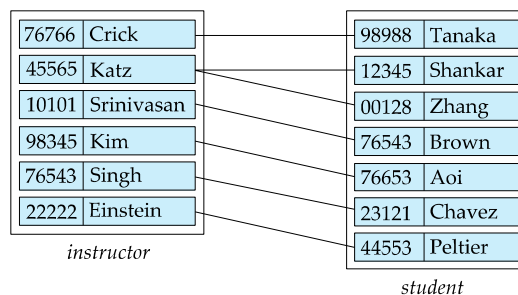


C'est quoi une instance de relation (relationship instance)??

MCD - Relationship Sets

■ L'association (ou relation) / Relationship

- Une relation est une **association entre plusieurs entités**.
 - **C'est quoi une instance de relation (relationship instance)??**
- Une instance de relation dans un schéma E-A / E-R représente une association entre les entités nommées dans le monde réel en cours de modélisation.
 - Voici une illustration:

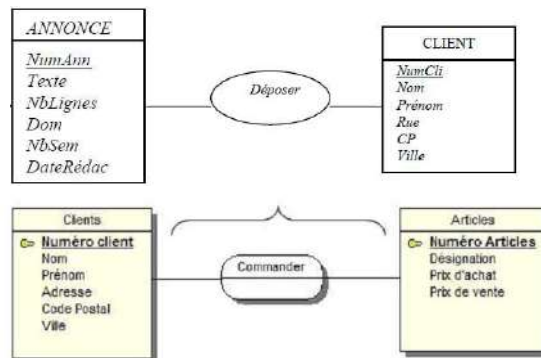


Ensemble de relations **advisor**

■ L'association (ou relation)

- C'est *un lien sémantique entre* plusieurs entités.
- Une classe de relation:
 - Contient donc toutes les relations de **même type**.
 - Peut lier **plus de deux** classes d'entité.

□ Exemple:



Notation E-A
Merise/MCD

PFM

Projet de Fin de Module –

Cahier de charge (propositions):

Considérons les deux ensembles d'entités étudiant et section, où section désigne une offre de cours.

Nous pouvons définir l'ensemble de relations pour désigner l'association entre un étudiant et une section dans laquelle cet étudiant est inscrit.

Bien que dans les exemples précédents chaque ensemble de relations soit une association entre deux ensembles d'entités, en général un ensemble de relations peut désigner l'association de plus de deux ensembles d'entités.

■ L'Association (merise E-A) / Relationship (E-R)

- La fonction qu'une entité joue dans une relation est appelée **rôle** de cette entité.
- Étant donné que les ensembles d'entités participant à un ensemble de relations sont généralement **distincts**, les rôles **sont implicites** et ne **sont généralement pas spécifiés**.
 - Cependant, ils sont utiles lorsque **le sens d'une relation** doit être clarifié.
- C'est le cas lorsque les ensembles d'entités d'un ensemble de relations ne sont pas distincts ;
 - c'est-à-dire que le même ensemble d'entités participe à un ensemble de relations plus d'une fois, **dans des rôles différents**.
- Dans ce type d'ensemble de relations, parfois appelé ensemble de relations **récurives**, des noms de rôle explicites sont nécessaires pour spécifier comment une entité participe à une instance de relation.

■ L'association (ou relation) / Relationship / Types de relations

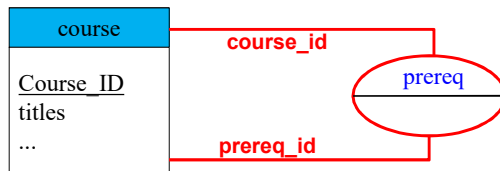
- **recursive relationship**
 - La même entité participe plus d'une fois à différents rôles.
 - Les relations récurives sont aussi parfois appelées **relations unaires**.
- Exemple:
- considérons d'ensemble d'entités **cours**.
 - Pour décrire la situation où un cours (C2) est un prérequis pour un autre cours (C1), nous avons un ensemble de **relations prereq** qui est modélisé par des paires ordonnées d'entités de cours.
 - Le premier cours d'un couple joue le rôle du cours C1, tandis que le second joue le rôle du cours prerequis C2.
 - Ainsi, toutes les relations de prereq sont caractérisées par des couples (C1, C2) ; Les paires (C2, C1) sont exclues.

■ L'association (ou relation) / Relationship / Types de relations

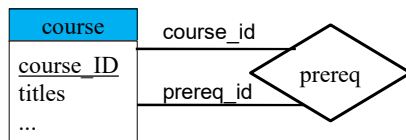
□ recursive relationship

□ Exemple:

- Schématisation montre les indicateurs de rôle course_id et prereq_id entre l'ensemble d'entités de cours et l'ensemble de relations prereq.



Notation
Merise/MCD

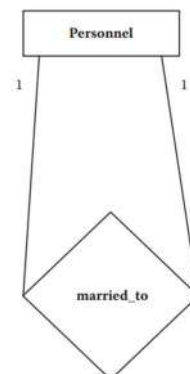
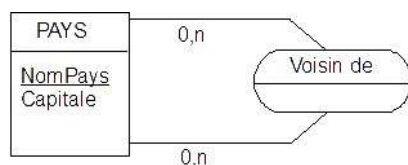


Notation
Entity-Relationship
Model

■ L'association (ou relation) / Relationship / Types de relations

□ recursive relationship

- Une classe de relation **récursive** (ou **réflexive**) relie la même classe d'entité.



Cahier de charge pour le cas d'étude:
Un module à des éléments module prérequis (voir le CNPN-2024
et les descriptifs des filières de l'ENSA)

■ L'association (ou relation) / Relationship / Types de relations

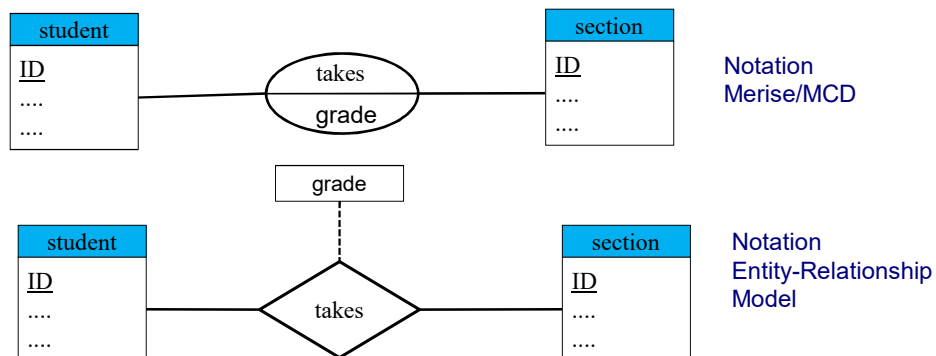
□ attributs descriptifs:

- **Attribut descriptif:** propriété ou caractéristique d'une relation (et non pas d'une entité).

Exemple:

□ considérons l'ensemble de relations qui relie les ensembles d'entités student et section.

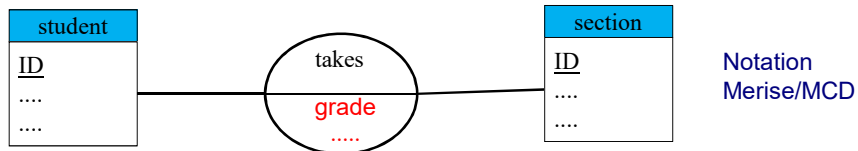
- Nous pouvons souhaiter stocker une note (grade) d'attribut descriptif avec la relation pour enregistrer la note qu'un étudiant a reçue dans une offre de cours.



■ L'association (ou relation) / Relationship / Types de relations

- attributs descriptifs (suite):

Exemple:



■ Note/Règle:

- Un ensemble de relations peut avoir **plusieurs attributs descriptifs**

■ Les cardinalités - Mapping Cardinalities

- Elles expriment le **nombre de fois** ou **l'occurrence** d'une entité participe aux occurrences de la relation.

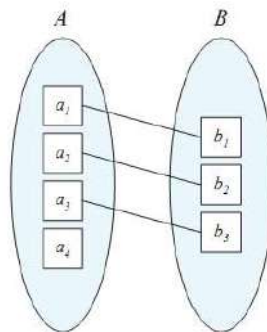
□ Exemple précédent:

- Combien de fois au **minimum** un client peut-il commander un article ?
- Combien de fois au **maximum** un client peut-il commander un article ?
 - nous pouvons répondre qu'un client, pour être client, doit commander au moins un article.
 - nous pouvons répondre qu'un client peut commander plusieurs articles.

■ Les cardinalités - Mapping Cardinalities

- Pour un ensemble de relations binaires R entre les ensembles d'entités A et B , la cardinalité de mappage doit être l'une des suivantes :

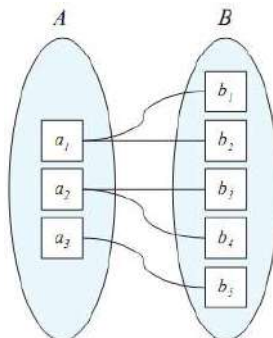
- **One-to-one / Un à un:** Une entité dans A est associée à **au plus** une entité dans B , et une entité dans B est associée à **au plus** une entité dans A .



■ Les cardinalités - Mapping Cardinalities

- Pour un ensemble de relations/associations binaires R entre les ensembles d'entités A et B , la cardinalité de mappage doit être l'une des cas suivantes :

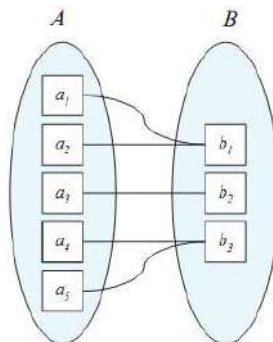
- **One-to-many / Un à plusieurs:** Une entité dans A est associée à **n'importe quel nombre** (zéro ou plus) d'entités dans B . Une entité dans B , cependant, peut être associée à **au plus** une entité dans A .



■ Les cardinalités - Mapping Cardinalities

- Pour un ensemble de relations binaires R entre les ensembles d'entités A et B , la cardinalité de mappage doit être l'une des suivantes :

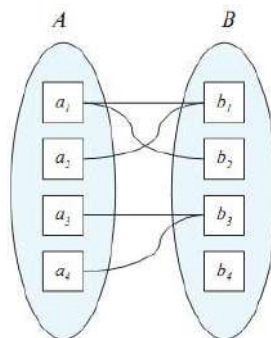
- **Many-to-one / Plusieurs à un:** Une entité dans A est associée à **au plus** une entité dans B . Une entité dans B , cependant, peut être associée à **n'importe quel nombre** (zéro ou plus) d'entités dans A .



■ Les cardinalités - Mapping Cardinalities

- Pour un ensemble de relations binaires R entre les ensembles d'entités A et B , la cardinalité de mappage doit être l'une des suivantes :

- **Many-to-Many / Plusieurs à Plusieurs:** Une entité dans A est associée à **n'importe quel nombre** (zéro ou plus) d'entités dans B , et une entité dans B est associée à **n'importe quel nombre** (zéro ou plus) d'entités dans A .



■ Totale et Partielle Participation

- Concerne les relations entre entités dans un schéma conceptuel.

■ Participation totale :

- chaque occurrence de l'entité liée doit être associée à **au moins une** occurrence de l'entité parente.

■ Participation partielle :

- dans une relation avec une participation partielle, une occurrence de l'entité liée **peut exister sans être associée** à une occurrence de l'entité parente. Cela signifie qu'il est possible d'avoir des occurrences de l'entité associée qui ne sont pas liées à l'entité parente.

- Ces concepts de participation totale et partielle aident à préciser les contraintes de la relation entre les entités dans un modèle conceptuel, ce qui permet de mieux définir les interactions et les dépendances entre les données.

■ Les cardinalités - Mapping Cardinalities

- Question : comment raisonner pour choisir les cardinalités entre entités? ...

■ Éléments de réponse

- Le choix dépend la situation réelle (le cas d'étude) à modéliser → la situation réelle que l'ensemble de relations modélise
 - Bien **lire le cahier de charge** ... À lire plusieurs fois.
 - Essayer de comprendre l'objectif global → et les sous-objectifs de votre SI

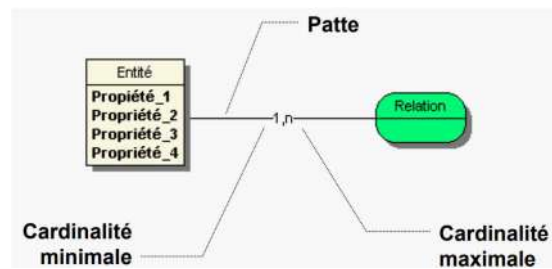
- Question: comment afficher les cardinalités sur un diagramme conceptuel?

- Cette participation s'analyse par rapport à une occurrence quelconque de l'entité type, et s'exprime par deux valeurs : la cardinalité **minimum** et la cardinalité **maximum**.

■ Les cardinalités - Mapping Cardinalities

□ Question: comment afficher les cardinalités sur un diagramme conceptuel?

■ Sur un MCD Merise :



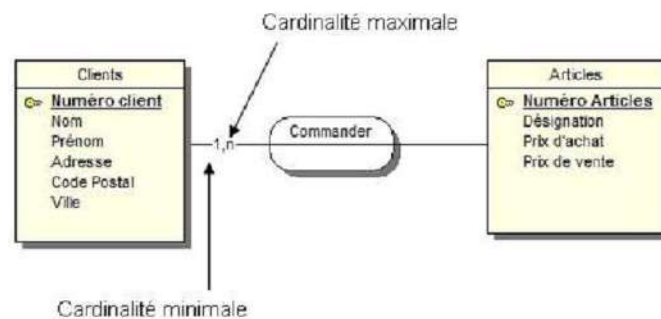
Analyser les situations suivantes et donner un exemple du monde réel??

Participation	Optionnelle	Obligatoire
Unique	0,1	1,1
Multiple	0,n	1,n

■ Les cardinalités - Mapping Cardinalities

□ Question: comment afficher les cardinalités sur un diagramme conceptuel?

■ Sur un MCD Merise :

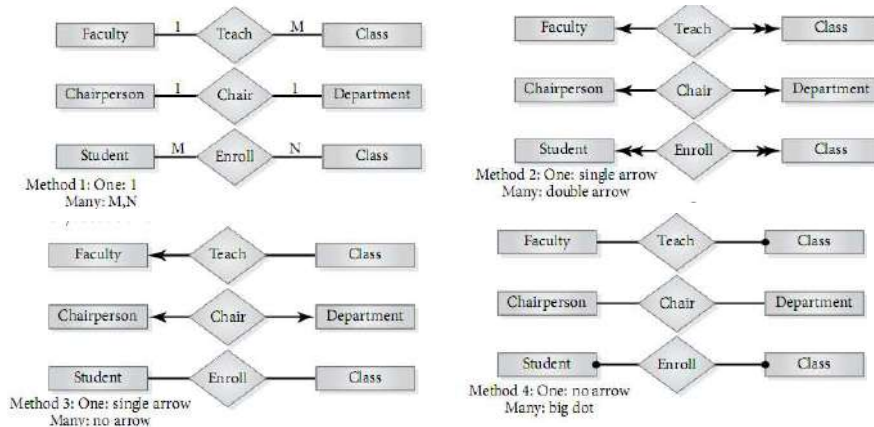


un client peut commander un ou plusieurs articles.

■ Les cardinalités - Mapping Cardinalities

□ Question: comment afficher les cardinalités sur un diagramme conceptuel?

■ Sur un le model E-R:

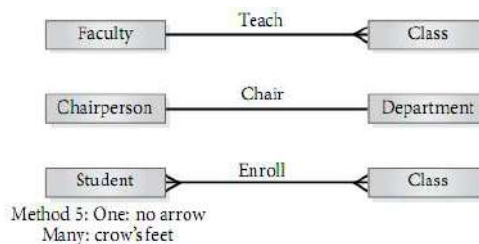


■ Les cardinalités - Mapping Cardinalities

□ Question: comment afficher les cardinalités sur un diagramme conceptuel?

■ Notation utilisée par SQL Developer **Data Modeler**

□ Modélisation de données avec Oracle SQL Developer (<https://www.oracle.com/fr/database/sqldeveloper/technologies/sql-data-modeler/>)



■ Les cardinalités - Mapping Cardinalities

□ Entité faible:

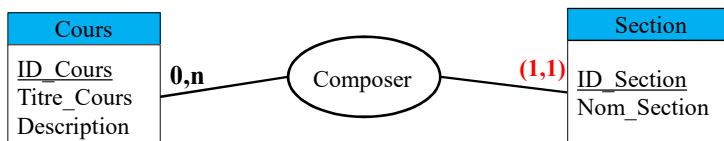
- Son identifiant est relatif
- Il faudra donc mettre entre parenthèses la cardinalité 1,1 pour indiquer que l'identifiant de l'entité concernée est relatif à l'autre entité en association.

□ Exemple d'entité faible entre les entités "Cours" et "Section" :

- Entité forte :
 - Cours : (ID_Cours, Titre_Cours, Description)
- Entité faible :
 - Section : (ID_Section, Nom_Section, ...)
- Synthèse:
 - Chaque section est associée à un cours spécifique.
 - une section ne peut pas être identifiée de manière unique sans référence au cours auquel elle est associée.
 - Par conséquent, l'entité "Section" est une entité faible qui dépend de l'entité "Cours" pour être identifiée.
- Dans un modèle conceptuel de données MERISE, vous auriez une relation de type "composition" entre l'entité forte "Cours" et l'entité faible "Section". Cela signifie qu'un cours peut être composé de plusieurs sections, mais chaque section est associée à un seul cours.

■ Les cardinalités - Mapping Cardinalities

□ Entité faible:



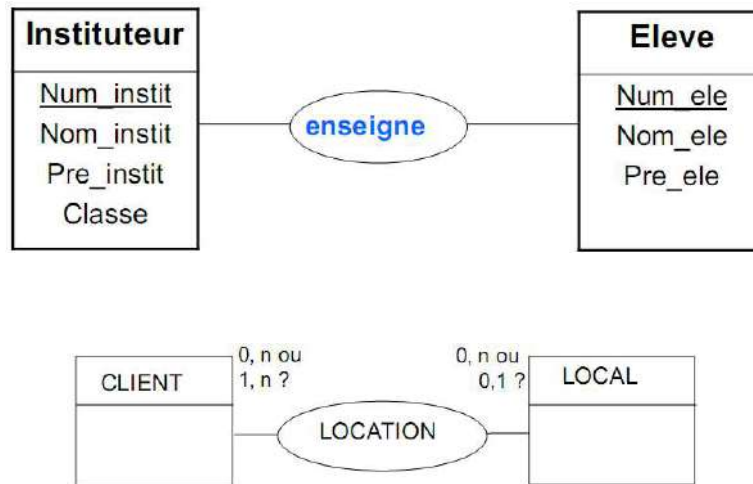
Notation
Merise/MCD

Une instance de cette relation

<u>ID_Cours</u>	Titre_Cours	Description
1	Mathématiques	Cours avancé sur les mathématiques
2	Physique	Cours élémentaire sur la physique

<u>ID_Section</u>	<u>ID_Cours</u>	Nom_Section
101	1	Groupe A
102	1	Groupe B
201	2	Groupe A
101	2	Groupe A

□ Proposer des cardinalités pour l'exemple suivant?



□ Proposer des cardinalités pour l'exemple suivant?

Un client peut il avoir 0 location ? Est-ce encore un client ?

Un local peut il être loué plusieurs fois ?

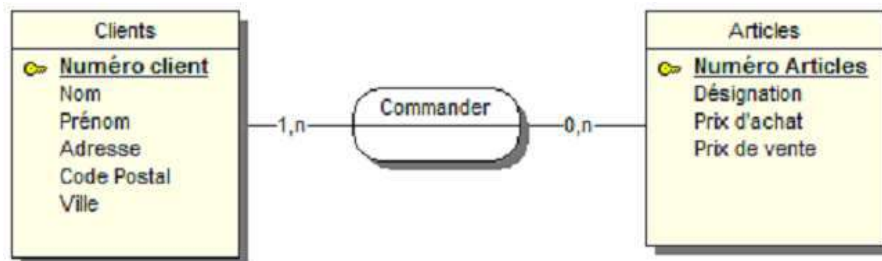
→ Non si la base représente une situation instantanée et si le local n'est pas partageable.

→ Oui si on gère un historique ou si le local est partageable.

Les cardinalités sont élément essentiel pour définir la sémantique des données, et non pas une « décoration » accessoire.

Derrière cette notion on trouvera des contrôles (par le SGBD ou les programmes).





un client peut commander un ou plusieurs articles.
Un article est commandé par 0 ou plusieurs clients

□ Remarque:

- Si le maximum est connu, il faut inscrire sa valeur.
- Par exemple, si dans les règles de gestion le client n'a le droit de commander qu'un maximum de 3 articles en tout et pour tout, dans ce cas là les cardinalités s'exprimeront de cette façon : 1,3.

■ Exercice:

- Modélisons le fait qu'une mère élève des enfants.

■ Contraintes de cardinalité sur les relations ternaires

- Si le maximum est connu, il faut inscrire sa valeur.
 - Par exemple, si dans les règles de gestion le client n'a le droit de commander qu'un maximum de 3 articles en tout et pour tout, dans ce cas les cardinalités s'exprimeront de cette façon : 1,3

■ Exercice:

- Modélisons le fait qu'une mère élève des enfants.

■ Les relations porteuses:

- Propriétés d'une relation
 - Une relation peut être dotée de propriétés
 - Exemple:

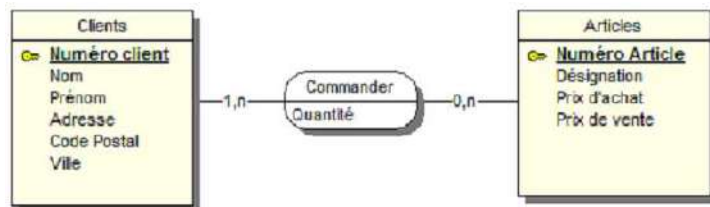


- Pourquoi est-ce qu'on ne peut pas associer la propriété Année à une des entités ?
- **Attention:** Cette propriété peut même devenir **une partie de l'identifiant**. Dans ce cas, elle doit être **soulignée**.



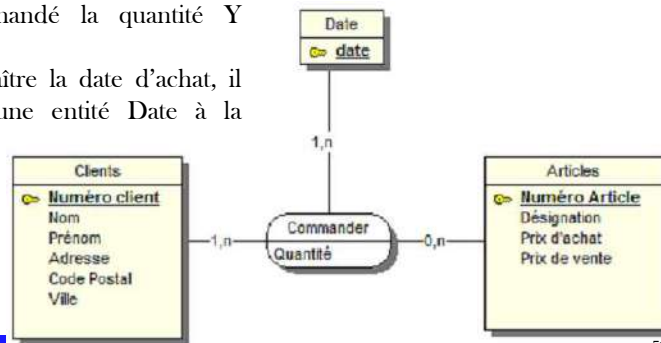
- L'identifiant de la **relation enseigner** Est : No_Matricule, Code_Classe, Année

■ Les relations porteuses:



Le **client** X a commandé la quantité Y d'**articles** Z.

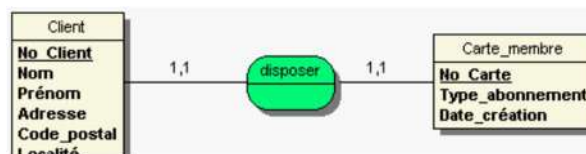
Si nous désirons connaître la date d'achat, il nous suffit de créer une entité Date à la relation **Commander**.



■ L'association (ou relation) / Relationship / Types de relations:

□ La dépendance d'une relation

- On dit qu'une entité est **indépendante** par rapport à **une relation** lorsque sa **cardinalité minimale** vaut 0, et **dépendante** par rapport à une relation lorsque sa **cardinalité minimale** vaut 1.
- Remarque: Une relation ne peut pas être liée uniquement à des entités **dépendantes** ayant **en plus une cardinalité maximale de 1 !!!**
 - Exemple: mauvaise conception (**Faux modèle**)



Solution/correction: Dans ce cas il faut réunir les propriétés des deux entités dans une seule entité (entité forte point de vue conception)

■ L'association (ou relation) / Relationship / Types de relations

□ Règles:

- Les conceptions E-A / E-R complexes peuvent devoir être divisées en plusieurs diagrammes qui peuvent être situés dans différentes pages.
- Les ensembles de relations doivent être affichés à un seul emplacement, mais les ensembles d'entités peuvent être répétés à plusieurs emplacements.
- Les attributs d'un ensemble d'entités doivent être affichés dans la première occurrence.
- Les occurrences suivantes de l'ensemble d'entités doivent être affichées sans attributs, pour éviter la répétition des informations et la possibilité d'incohérence qui en résulte dans les attributs affichés dans différentes occurrences.
- Il est possible d'avoir plusieurs ensembles de relations impliquant **les mêmes ensembles d'entités**.

■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

- Les relations binaires - binary relationship
 - Un ensemble qui implique **deux ensembles d'entités**.
 - Remarque:
 - La plupart des ensembles de relations dans un SI sont binaires.
- Parfois, les ensembles de relations impliquent plus de deux ensembles d'entités :
 - Le nombre d'ensembles d'entités qui participent à un ensemble de relations est **le degré de l'ensemble de relations**.
 - Un ensemble de relations binaires → est de degré 2 ;
 - Un ensemble de relations ternaires → est de degré 3.
 - Un ensemble de relations n-aires → est de degré n.

■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

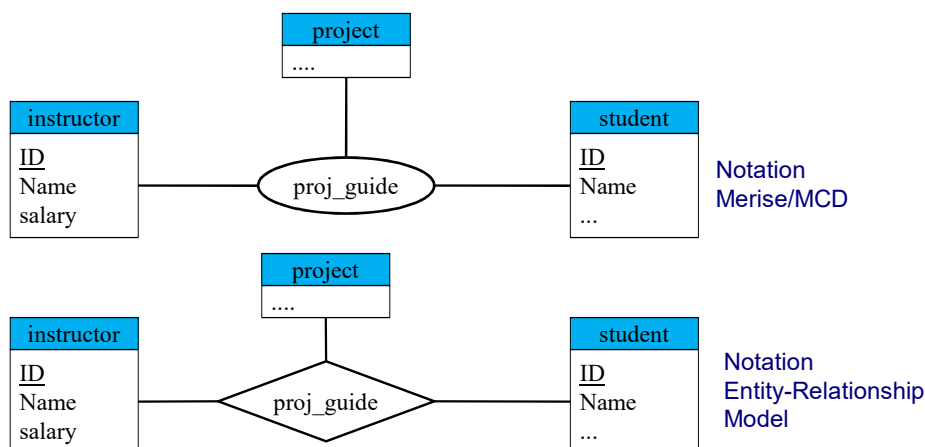
Exemple:

- Nous avons un ensemble d'entités qui représente tous les **projets** menés à l'université.
- Considérez les ensembles d'entités instructeur, étudiant et projet.
- Chaque projet peut avoir plusieurs étudiants associés et plusieurs instructeurs associés.
- De plus, chaque étudiant travaillant sur un projet doit avoir un instructeur associé qui supervise l'étudiant dans le projet.
- Modéliser le fait que:
 - un instructeur guide un étudiant dans un projet particulier.

■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

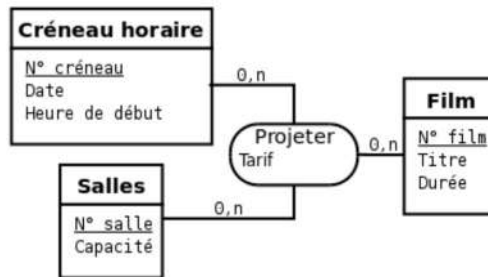
- Exemple : Les **étudiants** travaillent sur des **projets** de recherche sous la direction d'un **instructeur**.
- La relation **proj_guide** est une relation ternaire entre l'instructeur, l'étudiant et le projet.



■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

- Exemple : Un film est projeté dans une salle à une date donnée (créneau horaire)

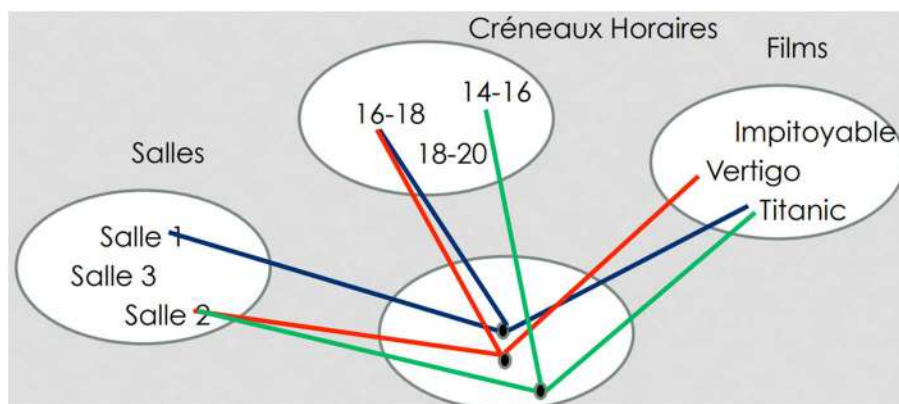
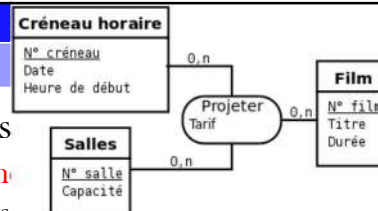


Question: Donner une instance de la relation projeter?

■ L'association (ou relation) / Relations

□ Degré d'une association / degree of the

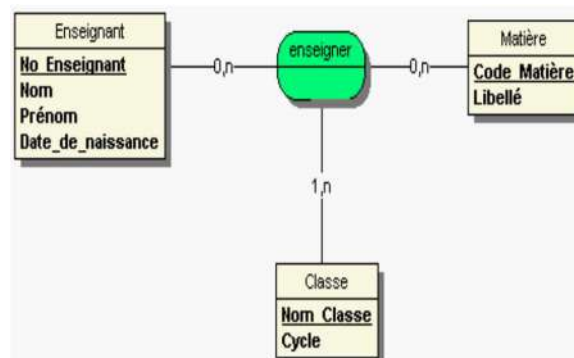
- Exemple : Un film est projeté dans une salle à une date donnée (créneau horaire)



■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

- Exercice TD: Question 1. Donner un cahier de charge du MCD suivant.



Question 2: Comment définir les cardinalités pour une relation ternaire?

■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

- Exercice TD: Question 1. Donner un cahier de charge du MCD suivant.

Question 2: Comment définir les cardinalités pour une relation ternaire?

Solution: Il faut **calculer les occurrences** pour **chaque patte**

Pour un enseignant X : combien de **couples (matière, classe)** pouvons nous créer?

card min = 0 car un enseignant peut ne pas enseigner du tout

card max = N car un enseignant peut enseigner plus

Pour une matière Y : combien de **couples(enseignant, classe)** pouvons nous créer?

card min = 0 car une matière peut ne pas être enseignée du tout

card max = N car une matière peut être enseignée plusieurs fois

Mersie – conception SIO -

MCD - Relationship Sets – Type de relations

- L'association (ou relation) / Relationship / Ternaire
 - Degré d'une association / degree of the relationship
 - Exercice TD: Question 1. Donner un cahier de charge du MCD suivant.

Question 2: Comment définir les cardinalités pour une relation ternaire?

Pour une classe Z: combien de couples (enseignant, matière) pouvons nous créer?

card min = 1 car une classe doit suivre au moins une matière

card max = N car une classe suivra plusieurs matières

A. BOULAALAM (USMBA/ENSA)
SI
61

Mersie – conception SIO -

MCD - Relationship Sets – Type de relations

- L'association (ou relation) / Relations
 - Degré d'une association / degree of the relationship
 - Ternaire ou Binaire?
 - Il est fortement conseillé de n'avoir que des relations/associations binaires. Le MCD Films avec des associations ternaires peut être transformé en MCD avec des associations binaires :

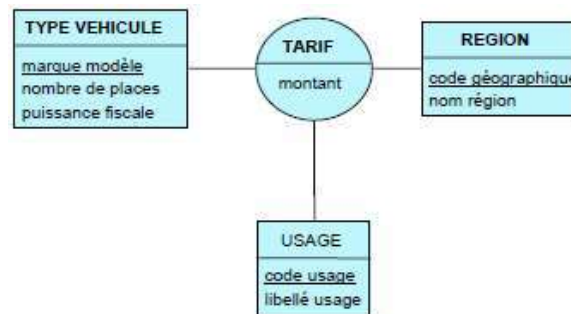
A. BOULAALAM (USMBA/ENSA)
SI
62

■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

■ Exercice TD:

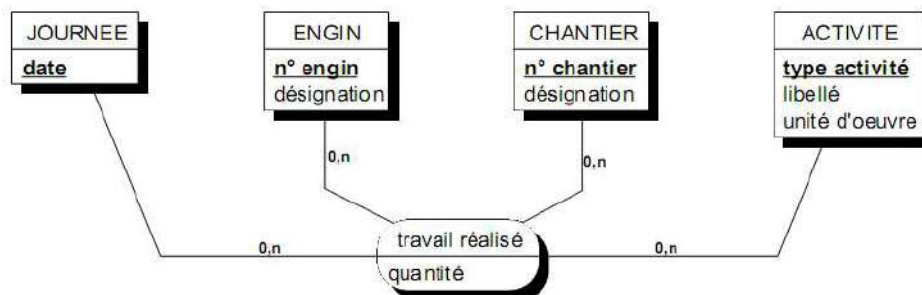
- Donner une description textuelle du diagramme suivant?
- Proposer le degré de participation (cardinalités) de chaque entité
- Donner une instance de la relation TARIF?



■ L'association (ou relation) / Relationship / Types de relations

□ Degré d'une association / degree of the relationship set.

■ Exercice TD:



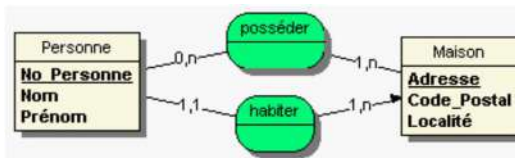
■ L'association (ou relation)/Relationship: Types de relations

□ Plusieurs relations types peuvent partager la même collection (ensemble d'entités):

- deux ou plusieurs entités types, des associations/relation de significations différentes peuvent exister.

■ Exemple:

- Faire une description du schéma suivant:



Une personne possède ou pas de maisons

Toute personne habite une et une maison

Une maison est possédée par au moins une personne

Une maison est habitée par au moins une personne

■ L'association (ou relation) / Relationship

□ Attributs complexes / Complex Attributes

- Attribut → il existe un ensemble de valeurs autorisées, appelé **domaine**, ou ensemble de valeurs, de cet attribut.

■ Exemple:

- attribut `course_id` c'est une chaîne de caractères d'une certaine longueur.
- attribut `semestre` → son domaine est peut être des chaînes de l'ensemble {automne, hiver, printemps, été}

- Un attribut, tel qu'utilisé dans le modèle E/A E/R, peut être caractérisé par les types d'attributs suivants:

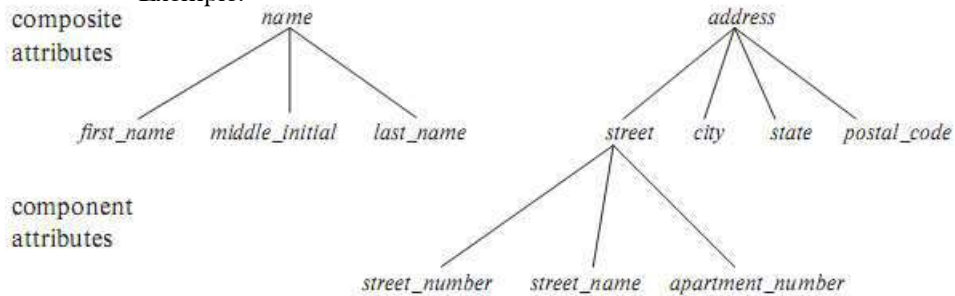
- Simple
- Composé/composite

■ L'association (ou relation) / Relationship

□ Attributs complexes / Complex Attributes

- Attributs Simples → c'est-à-dire qu'ils n'ont pas été divisés en sous-parties.
- Attributs composites → peuvent être divisés en sous-parties (c'est-à-dire d'autres attributs).

■ Exemple:



- Les attributs composites nous aident à regrouper les attributs liés, ce qui rend la modélisation plus propre.
- Notez également qu'un attribut composite peut apparaître sous forme de hiérarchie.

■ L'association (ou relation) / Relationship

□ Attributs à valeur unique.

- Les attributs de nos exemples ont tous une valeur unique pour une entité particulière.

- Un attribut fait référence à un seul valeur.

□ Attributs à valeurs multiples

■ Exemple:

- Supposons que nous ajoutons à l'ensemble d'entités instructeur un attribut de numéro de téléphone. Un instructeur peut avoir zéro, un ou plusieurs numéros de téléphone, et différents instructeurs peuvent avoir différents numéros de téléphone.
- Ce type d'attribut est dit **multivalué**.

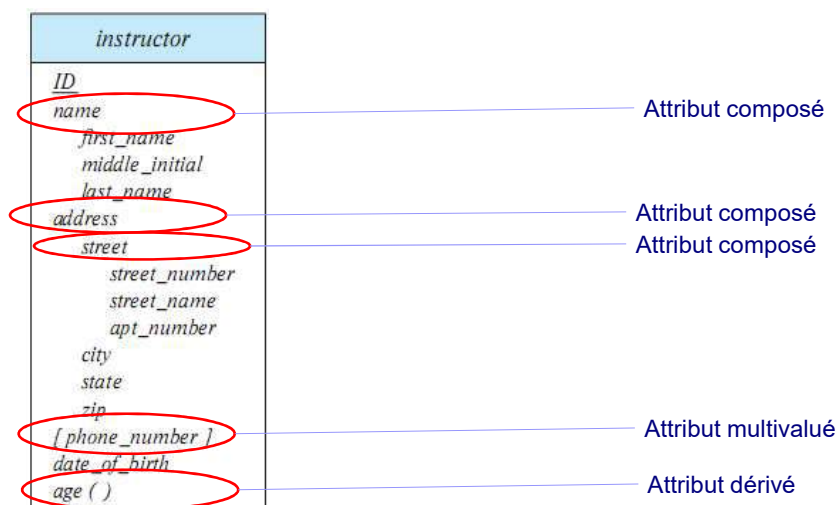
■ L'association (ou relation) / Relationship

□ Attributs dérivés.

- La valeur de ce type d'attribut peut être dérivée des valeurs d'autres attributs ou entités associés → calculé (Dans le DD Élémentaire vs Calculé)
- Par exemple:
 - disons que l'ensemble d'entités instructeur a un attribut *students_advised* → qui représente le nombre d'étudiants supervisés par un instructeur.
 - Nous pouvons déduire la valeur de cet attribut en comptant le nombre d'entités étudiantes associées à cet instructeur.
- Par exemple: l'ensemble d'entités instructeur ait un attribut âge qui indique l'âge de l'instructeur. Si l'ensemble d'entités instructeur a également un attribut date de naissance, nous pouvons calculer l'âge à partir de la date de naissance et de la date actuelle → Ainsi, l'âge est un attribut dérivé.
- Dans ce cas, la date de naissance peut être appelée **attribut de base** ou **attribut stocké**.
- La valeur d'un attribut dérivé n'est pas stockée mais il est calculée si nécessaire.

■ L'association (ou relation) / Relationship

□ composite, multivalued, and derived attributes.



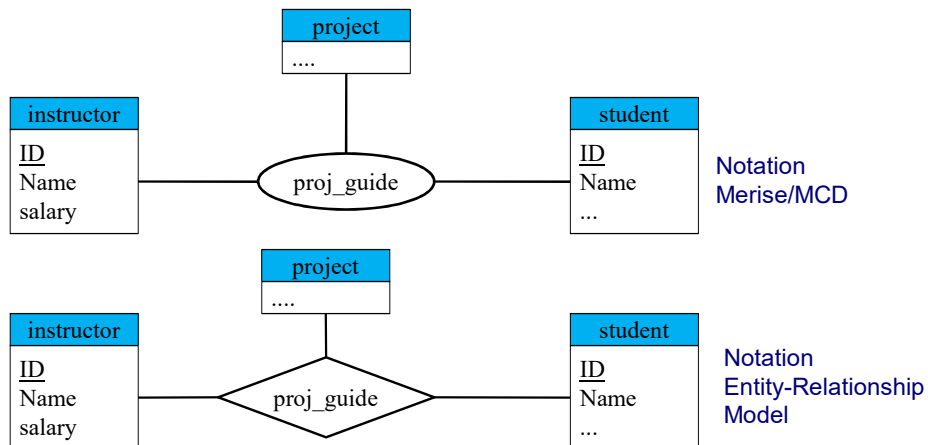
Cahier de charge pour le cas d'étude:
Dans la modélisation proposée, déterminer et discuter l'ensemble des attributs proposés.

■ L'association (ou relation) / Relationship / Types de relations

□ Exercice TD:

■ Exemple : Les **étudiants** travaillent sur des **projets** de recherche sous la direction d'un **instructeur**.

□ La relation **proj_guide** est une relation ternaire entre l'instructeur, l'étudiant et le projet.



Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 1

□ Deux entités qui doivent être reliées entre elles le seront par le biais d'une **relation/association**

The diagram illustrates Rule 1. The top part shows two entities, 'CLIENT' and 'FACTURE', connected by a simple line. This is labeled 'À ne pas faire' (Do not do). The bottom part shows the same two entities connected by an association named 'Reçoit'. The cardinalities are '1,n' for 'CLIENT' and '1,1' for 'FACTURE'. This is labeled 'OK'.

A. BOULAALAM (USMBA/ENSA) SI 73

Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 2

□ Deux relations/associations ne peuvent jamais être **directement reliées** entre elles

The diagram illustrates Rule 2. The top part shows two entities, 'PERSONNE' and 'IMMEUBLE', connected by two associations, 'Gère' and 'Possède'. These two associations are connected to each other. This is labeled 'À ne pas faire' (Do not do). The bottom part shows the same two entities connected by two separate associations, 'Gère' and 'Possède'. The cardinalities are '0,n' for 'PERSONNE' and '1,1' for 'IMMEUBLE'. This is labeled 'OK'.

A. BOULAALAM (USMBA/ENSA) SI 74

Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 3

□ Le nom de la relation doit être significatif:

- représenter d'une manière concrète et significative l'information que l'on veut obtenir

The diagram illustrates Rule 3: The name of the relation must be significant. It shows two examples of relationships between PERSONNE and IMMEUBLE. The top example, labeled 'À ne pas faire', shows a relationship named 'Concerne'. The bottom example, labeled 'OK', shows a relationship named 'Gère' with cardinalities 0,n on the PERSONNE side and 1,1 on the IMMEUBLE side.

A. BOULAALAM (USMBA/ENSA) SI 75

Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 4

□ Un attribut est unique à une entité ou à une relation

The diagram illustrates Rule 4: An attribute is unique to an entity or a relation. It shows two examples of relationships between PROFESSEUR and COURS. The top example, labeled 'À ne pas faire', shows a relationship named 'Peut donner' with cardinalities 0,n on both sides. The bottom example, labeled 'OK', shows a relationship named 'Peut donner' with cardinalities 0,n on both sides. The top example also shows attributes *NoProfesseur, NomProfesseur, PrénomProfesseur, Adresse for PROFESSEUR and *NoCours, Description, NomProfesseur for COURS. The bottom example shows attributes *NoProfesseur, NomProfesseur, PrénomProfesseur, Adresse for PROFESSEUR and *NoCours, Description for COURS.

A. BOULAALAM (USMBA/ENSA) SI 76

Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 5

□ Les entités et les relations ne doivent contenir que des données élémentaires:

- donc ne pas contenir des résultat de calcul/traitement

A. BOULALAM (USMBA/ENSA) SI 77

Mersie – conception SIO -
MCD – Quelques règles de conception

■ Règle 6

□ Pour une occurrence donnée:

- une seule valeur doit être attribuée à chaque attribut de l'entité ou de la relation

A. BOULALAM (USMBA/ENSA) SI 78

Mersie – conception SIO -

MCD – Quelques règles de conception

■ **Règle 7**

□ Toute relation binaire avec cardinalité (1,1) **ne peut être porteuse de propriété**.

■ En effet une telle propriété **migre** alors **obligatoirement** dans l'entité portant cette cardinalité (1,1).

incorrect

correct

A. BOULAALAM (USMBA/ENSA) SI 79

PFM

Cahier de charge – suite

Projet de Fin de Module –

Les étapes de conception à faire: (C'est une proposition ☺)

Objectif : montrer comment un tel modèle peut être utilisé comme outil d'analyse et de spécification dans le cadre d'un cahier des charges réel du SI de l'ENSA.

Les six tâches suivantes sont **imbriquées** et **cycliques**:

1. **Identifier les entités**. En se basant sur le DD ...
2. **Identifier les attributs (propriétés)**.
3. **Décrire les entités et attributs**, de préférence dans le dictionnaire de données.
4. **Identifier les relations**.
5. **Définir les cardinalités**.
6. **Vérifier la cohérence** de l'ensemble et remonter à la première étape si nécessaire ☺

A. BOULAALAM (USMBA/ENSA) SI 80

MPD

Mersie – conception SIO -

MPD- Modèle Physique des Données

■ Définition:

- ☐ Construire le Modèle Physique des Données consiste à transformer le Modèle Logique des Données en une suite de relations.
- ☐ Cette étape finalise le processus de traitement des données.
- ☐ L'implémentation des bases de données peut être réalisée de façon optimale.

■ Passer du modèle logique de données au modèle physique des données ne présente aucune difficulté:

On abandonne juste la représentation graphique pour une représentation plus linéaire.

■ Exercice TD

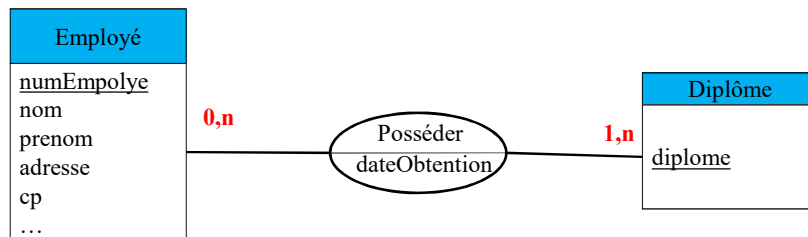
- ☐ Cahier de charge (Extrait)
 - Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
 - Un employé possède des diplômes dans des dates différentes. Un diplôme est identifié par son intitulé.
 - Question 1: Proposer un MCD
 - Question 2: Donner le MLD correspondant.
 - Question 3: Donner le MPD correspondant.

■ Exercice TD

□ Cahier de charge (Extrait)

- Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
- Un employé possède des diplômes. Un diplôme est identifié par son intitulé.

1. **Question 1: Proposer un MCD**
2. Question 2: Donner le MLD correspondant.
3. Question 3: Donner le MPD correspondant.

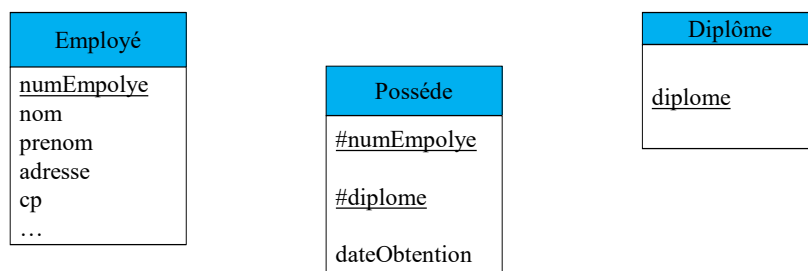


■ Exercice TD

□ Cahier de charge (Extrait)

- Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
- Un employé possède des diplômes. Un diplôme est identifié par son intitulé.

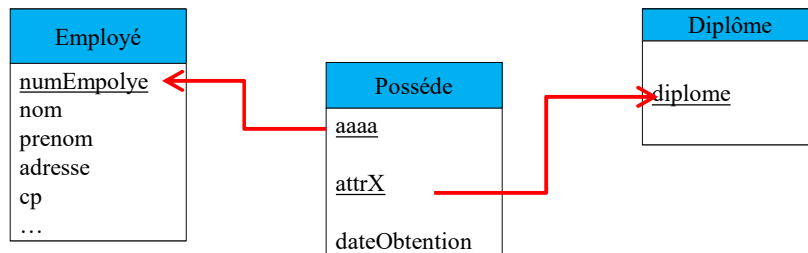
1. Question 1: Proposer un MCD
2. **Question 2: Donner le MLD correspondant.**
3. Question 3: Donner le MPD correspondant.



■ Exercice TD

□ Cahier de charge (Extrait)

- Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
 - Un employé possède des diplômes. Un diplôme est identifié par son intitulé.
1. Question 1: Proposer un MCD
 2. **Question 2: Donner le MLD correspondant. (Autre convention)**
 3. Question 3: Donner le MPD correspondant.



■ Exercice TD

□ Cahier de charge (Extrait)

- Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
 - Un employé possède des diplômes. Un diplôme est identifié par son intitulé.
1. Question 1: Proposer un MCD
 2. Question 2: Donner le MLD correspondant. (Autre convention)
 3. **Question 3: Donner le MPD correspondant.**

Employé (numEmpolye, nom, prenom, adresse, cp, ...)

Possède (#numEmpolye #diplome, dateObtention)

Diplôme (diplome)

■ Exercice TD

□ Cahier de charge (Extrait)

- Dans une entreprise un employé a les informations suivantes: un nom et prénom, une adresse, un code postale un vile et un téléphone. Pour bien identifier les employés, un numéro employé est attribué à chaque employé.
 - Un employé possède des diplômes. Un diplôme est identifié par son intitulé.
1. Question 1: Proposer un MCD
 2. Question 2: Donner le MLD correspondant. (Autre convention)
 3. **Question 3: Donner le MPD correspondant. (autre convention)**

Employé (numEmpolve, nom, prenom, adresse, cp, ...)

Possède (#att1 #att2, dateObtention)

Diplôme (diplome)

The Relational Model and FDs

■ INTRODUCTION

- Selon Edgar Codd pour la base de données **relationnelle** :
 - les données devaient être organisées dans de simples tables bidimensionnels non ordonnés.
 - En organisant correctement les données de cette manière, nous pouvons poser des questions sur le contenu de la base de données (générer des requêtes) de manière simple ;
 - nous ignorons essentiellement la façon dont les données sont réellement stockées.
- Discussion: l'idée de « table appropriée » ET dépendances fonctionnelles (FD):
 - Comme Codd l'a suggéré, nous ne créerons pas notre base de données avec des **enregistrements physiquement liés**, mais nous nous assurerons plutôt que nos données sont sous une **forme appropriée**.
 - La forme appropriée signifie que les données **sont normalisées**, et la normalisation est obtenue en **suivant la notion de DFs**

■ Exemple

	column1	column2	column3	...
Exemple de relation/Table	row1	data-cell	data-cell	data-cell
	row2	data-cell	data-cell	data-cell
	row3	data-cell	data-cell	data-cell

	Name	Phone_no	Address
Exemple: La relation/Table	Jones	222-3333	123 4th St
CUSTOMER	Smith	333-2154	55 Main St
	Adams	555-8888	3145 Euclid Ct

Souvent, lors de la discussion d'une ligne dans un tableau, elle est notée comme suit :

<Jones, 222-3333, 123 4th St>

Les cellules de données contiennent des faits, par exemple:

vous avez un client nommé Smith, l'adresse de Smith est 55 Main Street, etc.

	Name	Phone_no	Address
Question: Correcte ou non ??	Jones	222-3333	123 4th St
	55 Main St	Smith	333-2154
	Adams 3145	Euclid Ct	555-8888

■ Remarque

□ Dans les bases de données relationnelles:

- la cellule de données est supposée être **atomique**.
 - La caractéristique d'atomicité signifie que la cellule contient un fait et un seul fait.

■ Exemple:

- Si Adams avait deux numéros de téléphone et s'ils étaient entrés sur la même ligne, ce ne serait pas une table de données valide pour une base de données relationnelle. Vous auriez à concevoir les tables d'une autre manière

Question: Correcte ou non ??

Name	Phone_no	Address
Jones	222-3333	123 4th St
Smith	333-2154	55 Main St
Adams	555-8888, 555-8889	3145 Euclid Ct

■ 1FN

□ Exemple (suite)

- Cet arrangement de données avec une cellule non atomique n'est pas une table, contrairement à l'exemple précédent avec toutes les cellules atomiques.
- Dans la terminologie de base de données de Codd, si vous avez une table (c'est-à-dire un arrangement bidimensionnel de données atomiques), vos données sont sous la première **forme normale (1NF)**.

□ Soit R une relation/table et A, B, C et D sont des attributs, Alors:

- $R(A, B, C, D) \rightarrow$ est en 1NF, alors que
- $R(A, B, C, \{D\}) \rightarrow$ n'est pas 1NF car $\{D\}$ représente un groupe répétitif.

■ DF - Dépendance fonctionnelle (rappel)

- Une DF est une relation d'un attribut dans une relation à un autre.
- Exemple:
 - nous pouvons dire que le numéro de sécurité sociale (SSN) définit ou identifie un nom.
 - Qu'est-ce que cela signifie? Cela signifie que si j'ai une base de données avec des SSN et des noms, et si je connais le SSN de quelqu'un, alors je peux trouver le nom de la personne.
 - De plus, puisque nous avons utilisé le mot définit, nous disons que pour chaque SSN, nous aurons **un et un seul** nom. Nous dirons que nous avons classé le nom comme **fonctionnellement dépendant** du SSN.
- C'est l'idée d'une clé primaire dont nous avons discuté précédemment.
- L'idée principale des DF est de trouver des clés primaires telles que toutes les données d'un enregistrement dépendent uniquement de la clé primaire.
- Dans une base de données, le concepteur fait des choix définissant les données avec des DF.
 - Rappelons qu'il est de la **responsabilité du concepteur** d'obtenir les informations FD de l'utilisateur.

■ DF - Dépendance fonctionnelle (rappel)

- Exemple:
 - supposons qu'une entreprise attribue à chaque employé un numéro d'employé unique.
 - Chaque employé a un numéro d'employé et un nom.
 - Les noms peuvent être les mêmes pour deux employés différents, mais pour deux employés, leurs numéros d'employés seront toujours différents et uniques parce que l'entreprise les a définis de cette façon. Il serait incohérent dans la base de données s'il y avait deux occurrences du même numéro d'employé avec des noms différents.
- Nous écrivons un FD avec une flèche comme ceci :
 - $SSN \rightarrow Name$
 - $EmpNo \rightarrow Name$
 - se lit « Empno définit le nom » ou « Empno implique le nom ».

■ NON-1NF À 1NF

□ Exemple:

CUSTOMER			
Customer_no	Name	Phone_no	Address
101	Jones	222-3333	123 4th St
102	Smith	333-2154	55 Main St
107	Adams	555-8888, 555-8889	3145 Euclid Ct

Avant de traiter le problème non-1NF, il est préférable de définir une clé primaire dans R si possible.

Supposons que la clé primaire soit l'attribut A (Customer_no),
Ensuite, la façon dont le groupe répétitif est géré passe par un processus appelé décomposition.

Le tableau d'origine, R(A, B, {C}, D), sera décomposé en deux tableaux :

R1(A, B, D) (la PK est A)

R2(A,C) (la PK est A et C)

■ NON-1NF À 1NF

□ Exemple:

CUSTOMER			
Customer_no	Name	Phone_no	Address
101	Jones	222-3333	123 4th St
102	Smith	333-2154	55 Main St
107	Adams	555-8888, 555-8889	3145 Euclid Ct



CUSTOMER1 (Customer_no, Name, Address)

CUSTOMER2 (Customer_no, Phone_no)

CUSTOMER1		
Customer_no	Name	Address
101	Jones	123 4th St
102	Smith	55 Main St
107	Adams	3145 Euclid Ct

CUSTOMER2	
Customer_no	Phone_no
101	222-3333
102	333-2154
107	555-8888
107	555-8889

■ NON-1NF À 1NF

□ Exemple:

CUSTOMER1		
Customer_no	Name	Address
101	Jones	123 4th St
102	Smith	55 Main St
107	Adams	3145 Euclid Ct

CUSTOMER2	
Customer_no	Phone_no
101	222-3333
102	333-2154
107	555-8888
107	555-8889

Les 2 tables sont en 1NF.

La clé primaire de la première table, Customer_no dans CUSTOMER1, est référencée dans la deuxième table. Customer_no dans CUSTOMER2 est appelé une clé étrangère car il fait référence à une clé primaire dans une autre table.

Méthode 1:

CUSTOMER1 (Customer_no, Name, Address)

CUSTOMER2 (#Customer_no, Phone_no)

Méthode 2:

CUSTOMER1 (Customer_no, Name, Address)

CUSTOMER2 (Customer_no, Phone_no)

■ NON-1NF À 1NF

□ Exemple: (Version 2)

- Supposons qu'il n'y ait pas de numéro de client, pas de clé évidente. Supposons que les données d'origine ressemblent à ceci :

CUSTOMER		
Name	Phone_no	Address
Jones	222-3333	123 4th St
Smith	333-2154	55 Main St
Adams	555-8888, 555-8889	3145 Euclid Ct

Discussion:

Ni le nom ni l'adresse ne seraient considérés comme un PK & nous avons un groupe répétitif.

Qu'est-ce qu'on fait?! ☺

Nous prenons les valeurs de groupe répétitives et les combinons avec tous les autres attributs et appelons la ligne entière une clé. Est-ce la meilleure clé que vous puissiez choisir ? Peut-être !

Ici, nous avons pris un arrangement de données non-1NF et en avons fait 1NF.

De plus, cette technique fonctionne toujours car vous vous retrouvez avec des données atomiques (1NF). → est transformé en une table comme celle-ci avec Name et Phone_no comme clé : CUSTOMER (Name, Phone_no, Address)

■ NON-1NF À 1NF

□ Exemple: (Version 2)

- Supposons qu'il n'y ait pas de numéro de client, pas de clé évidente. Supposons que les données d'origine ressemblent à ceci :

CUSTOMER		
Name	Phone_no	Address
Jones	222-3333	123 4th St
Smith	333-2154	55 Main St
Adams	555-8888, 555-8889	3145 Euclid Ct

Discussion:

De plus, cette technique fonctionne toujours car vous vous retrouvez avec des données atomiques (1NF). → CUSTOMER ([Name](#), [Phone_no](#), Address)

Name	Phone_no	Address
Jones	222-3333	123 4th St
Smith	333-2154	55 Main St
Adams	555-8888	3145 Euclid Ct
Adams	555-8889	3145 Euclid Ct

→ 1NF, des lignes uniques et une clé primaire.

→ Remarquez que dans le cas de CUSTOMER avec Customer_no qui vient d'être donné, il pourrait être résolu en 1NF comme ceci :

■ NON-1NF À 1NF

□ Exemple: (Version 2)

- Supposons qu'il n'y ait pas de numéro de client, pas de clé évidente. Supposons que les données d'origine ressemblent à ceci :

CUSTOMER		
Name	Phone_no	Address
Jones	222-3333	123 4th St
Smith	333-2154	55 Main St
Adams	555-8888, 555-8889	3145 Euclid Ct

→ 1NF, des lignes uniques et une clé primaire.

→ Remarquez que dans le cas de CUSTOMER avec Customer_no qui vient d'être donné, il pourrait être résolu en 1NF comme ceci :

CUSTOMER ([Customer_no](#), [Name](#), [Phone_no](#), Address, City)

Customer_no	Name	Phone_no	Address	City
101	Jones	222-3333	123 4th St	Pensacola
102	Smith	333-2154	55 Main St	Alpharetta
107	Adams	555-8888	3145 Euclid Ct	Two Egg
107	Adams	555-8889	3145 Euclid Ct	Two Egg



■ NON-1NF À 1NF - Exercice

□ Soit la table:

Commande (n° commande, date, n° client, nom, n° article, désignation, qté cmdée).

L'attribut nom est le nom du client et désignation est la désignation de l'article.

Question: Discuter la 1NF ?

■ NON-1NF À 1NF - Exercice

□ Soit la table:

Commande (n° commande, date, n° client, nom, n° article, désignation, qté cmdée).

L'attribut nom est le nom du client et désignation est la désignation de l'article.

Question: Discuter la 1NF ?

Commande

Etape 1: Discuter les DFs

n° commande

date

n° client

Nom

n° article

Désignation

qté cmdée

■ NON-1NF À 1NF - Exercice

□ Soit la table:

Commande (n° commande, date, n° client, nom, n° article, désignation, qté cmdée).

L'attribut nom est le nom du client et désignation est la désignation de l'article.

Question: Discuter la 1NF ?

Commande

n° commande

date

n° client

Nom

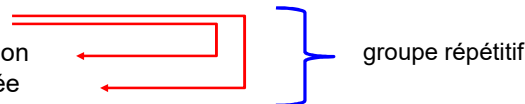
n° article

Désignation

qté cmdée

Etape 1: Discuter les DFs ... PK

Etape 2: Trouver les groupes répétitifs!



Etape 3:
Eliminer les
groupe répétitif

■ NON-1NF À 1NF - Exercice

□ Soit la table:

Commande (n° commande, date, n° client, nom, n° article, désignation, qté cmdée).

L'attribut nom est le nom du client et désignation est la désignation de l'article.

Question: Discuter la 1NF ?

Commande

n° commande

date

n° client

Nom

n° article

Désignation

qté cmdée

Tables en 1NF obtenues après élimination des groupements répétitifs :

table Commande

n° commande

(clé primaire)

date

n° client

nom

table Article_commande

n° commande

(clé primaire composée)

n° article

désignation

qté cmdée



■ 2NF - Deuxième Forme Normale

- ☐ Pour discuter la 2FN → La 1NF est vérifiée.
- ☐ A suivre ...

■ Exercices

Fin de partie