

RÉSUMÉ THÉORIQUE – FILIÈRE DÉVELOPPEMENT DIGITAL

M106 – Manipuler des bases de données

45 Heures

SOMMAIRE

1. **Concevoir une base de données**
   - Analyser le cahier de charges
   - Modéliser les données
   - Normaliser les données
2. **Préparer l'environnement**
   - Exploiter un outil de modélisation
   - Préparer le serveur MySQL
3. **Manipuler les données**
   - Créer une base de données
   - Réaliser des requêtes SQL
   - Administrer une base de données



MODALITÉS PÉDAGOGIQUES



1

LE GUIDE DE SOUTIEN

Il contient le résumé théorique et le manuel des travaux pratiques



2

LA VERSION PDF

Une version PDF est mise en ligne sur l'espace apprenant et formateur de la plateforme WebForce Life



3

DES CONTENUS TÉLÉCHARGEABLES

Les fiches de résumés ou des exercices sont téléchargeables sur WebForce Life



4

DU CONTENU INTERACTIF

Vous disposez de contenus interactifs sous forme d'exercices et de cours à utiliser sur WebForce Life

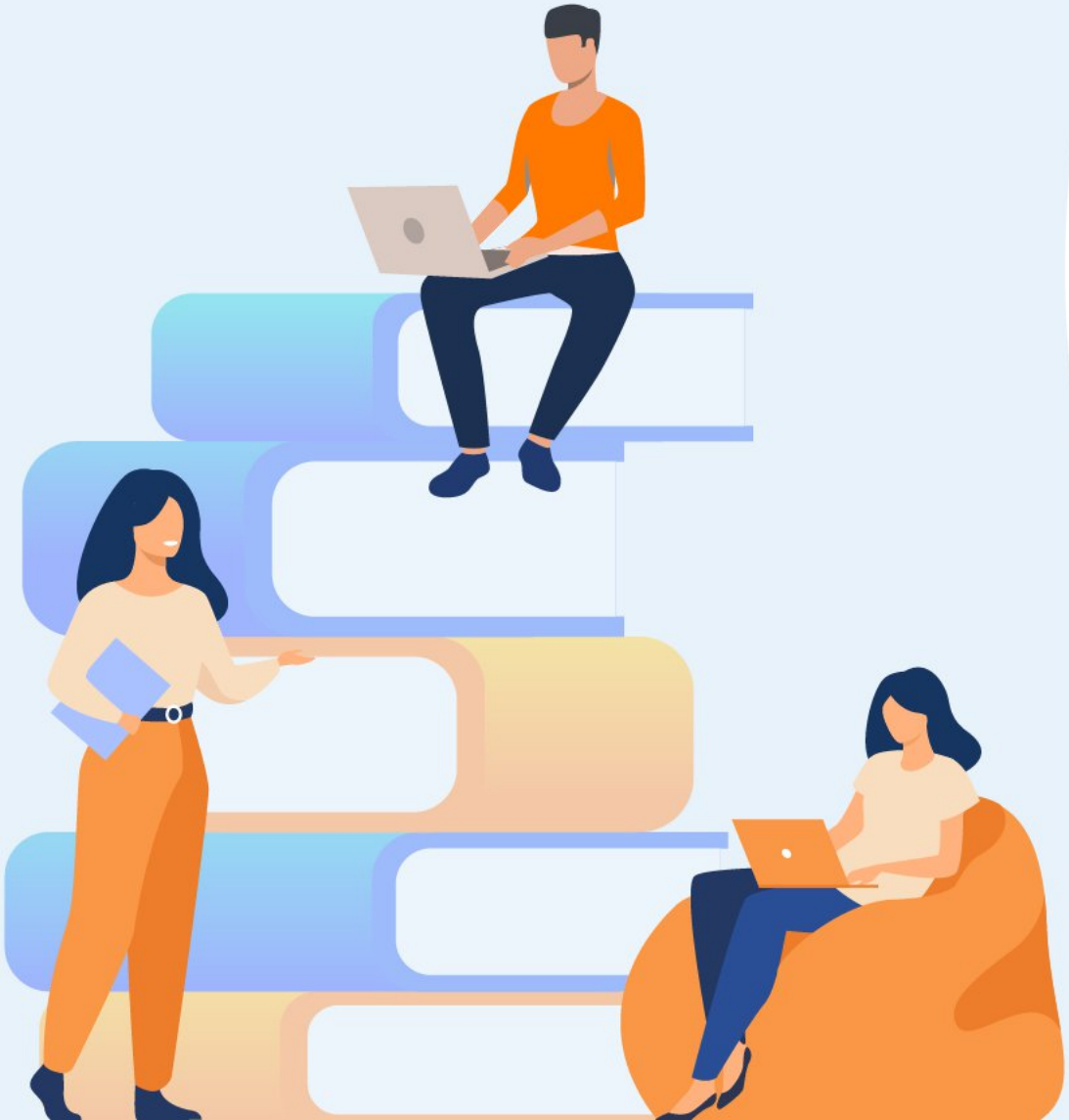


5

DES RESSOURCES EN LIGNES

Les ressources sont consultables en synchrone et en asynchrone pour s'adapter au rythme de l'apprentissage

Copyright - Tout droit réservé - OFPPT



PARTIE 1

Conception d’une base de données

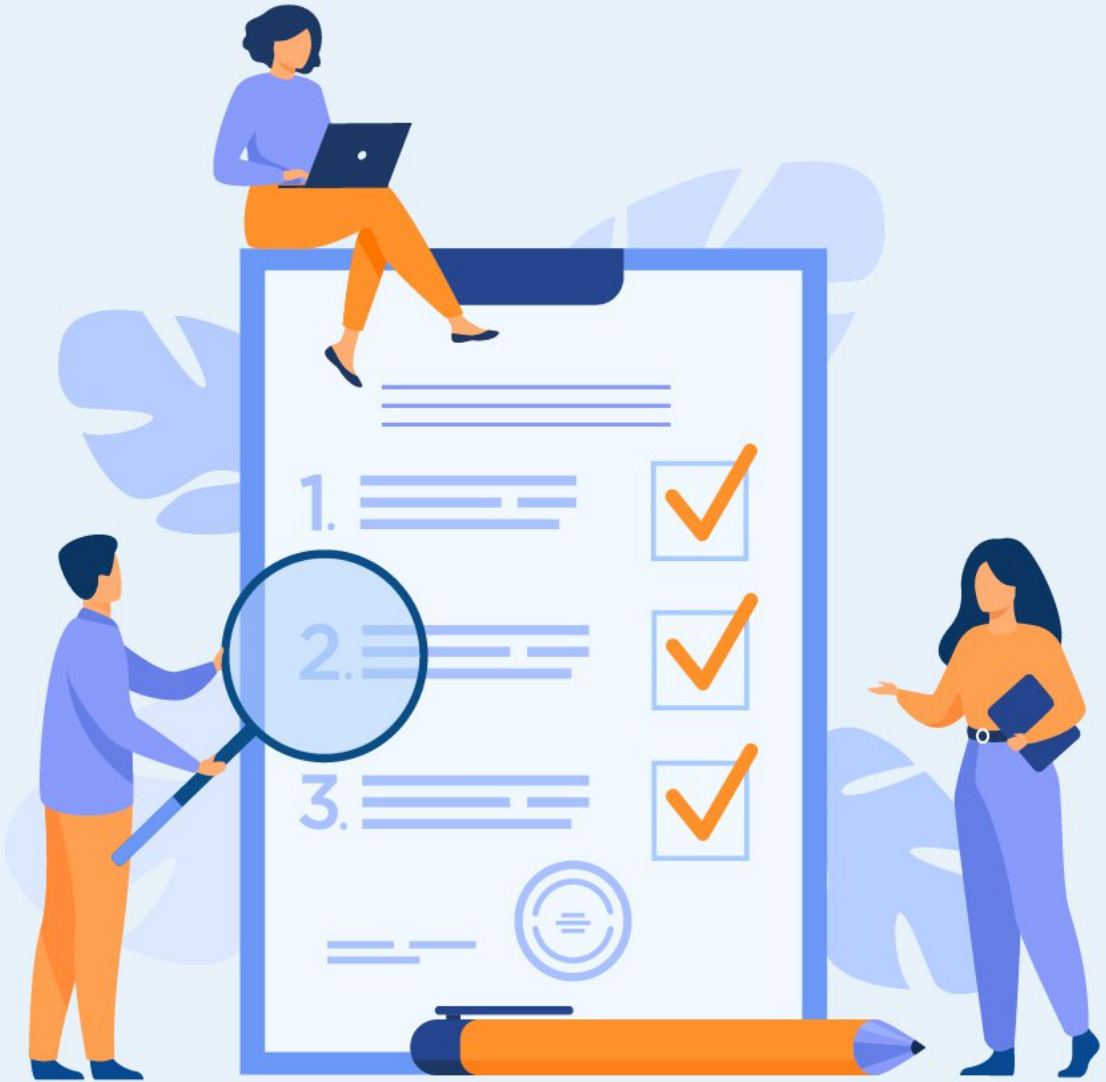
Dans ce module, vous allez :

- Analyser les données d’un cahier de charges
- Construire des modèles conceptuels (MCD) et logiques (MLD) de données
- Maitriser la normalisation
- Connaitre les règles de passage du MCD au MLD normalisé

17 Heures

CHAPITRE 1

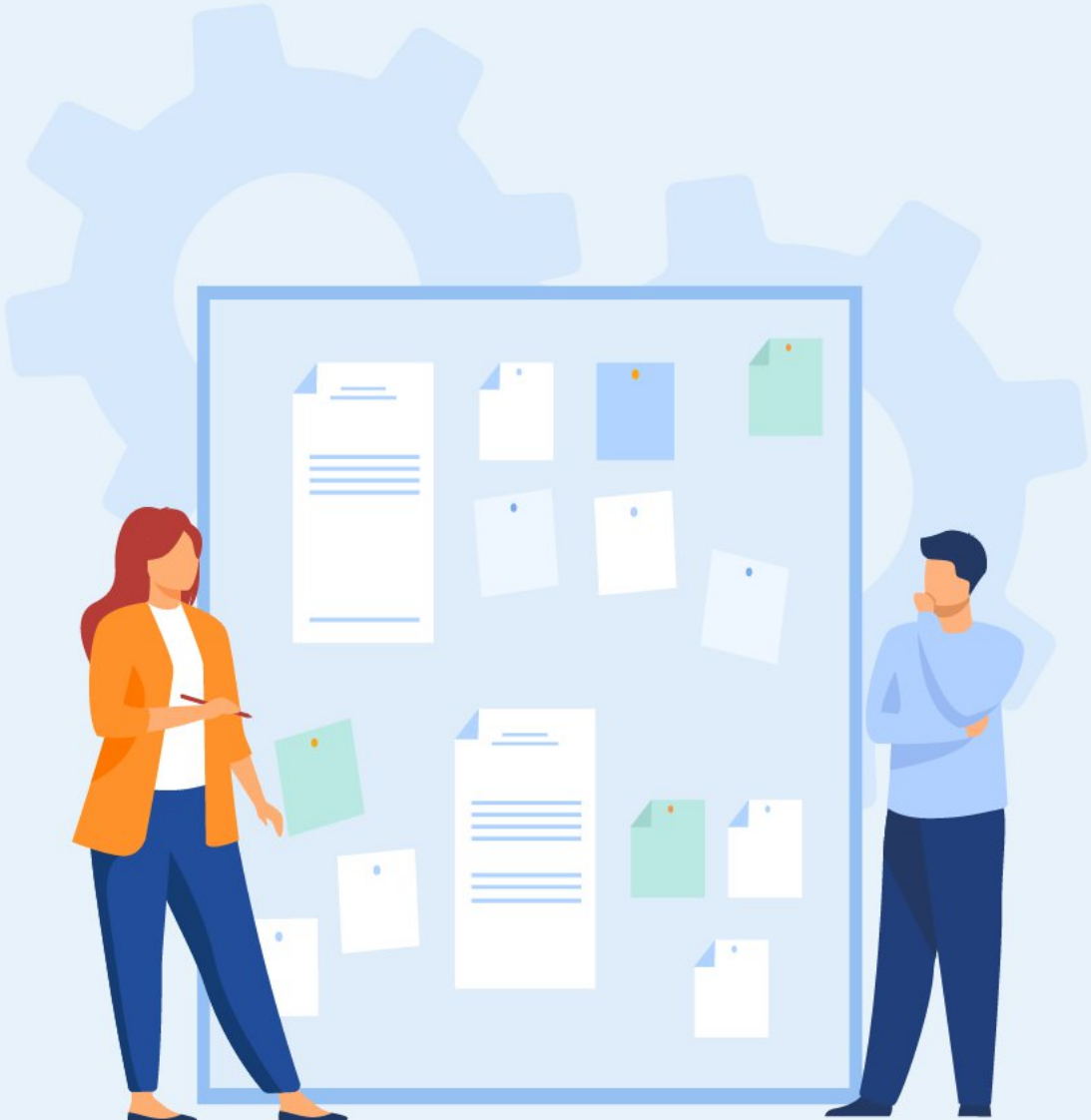
ANALYSE DU CAHIER DES CHARGES



Ce que vous allez apprendre dans ce chapitre :

- La lecture et l’analyse d’un cahier des charges
- L’identification des limites du projet
- L’ analyse des données et des traitements de la situation présentée

04 Heures



CHAPITRE 1

ANALYSE DU CAHIER DES CHARGES

1. **Lecture d’un cahier des charges**
2. Description des limites du projet
3. Analyse des données et des traitements de la situation présentée

PARTIE 1

Introduction

- Dans chaque organisation, Il y'a une quantité importante d'informations qui sont échangées afin d'assurer le bon fonctionnement de cette organisation ainsi que la communication avec son environnement. Dans le but d' être utilisables dans les activités opérationnelles quotidiennes ou encore dans la prise de décision. Ces informations doivent êtres bien organisées et stockées. Il est donc nécessaire pour chaque organisme d'avoir une structure fonctionnelle et technique de gestion de l'information.

Le système d'information

- Le système d'information représente l'ensemble des éléments participants au activités d'acquérir, de stocker, de traiter et de communiquer les informations au sein d'une organisation. Il se compose des acteurs suivants :
  - ✓ **Les individus :** En plus des spécialistes des Systèmes d'Information charges de la conception, la mise en œuvre et la gestion du système d'information, cette catégorie comprend aussi Les personnes qui utilisent ce dernier pour acquérir, communiquer, stocker ou traiter des informations.
  - ✓ **Le matériel :** Il s'agit de tout dispositif physique permettant d'émettre, manipuler ou stocker l'information.
  - ✓ **Les logiciels et les procédures :** Ce sont les programmes qui sont nécessaires au fonctionnement du Système d'Information ainsi que les procédures qui gèrent les traitements manuels et automatisés.
  - ✓ **Les données :** Elles constituent la matière première des traitements : saisies, déduites ou calculées.

- Dans le cadre d'un système d'information, un projet informatique a pour objectif de construire une application informatique (logiciel et base de données) qui va servir comme un support informatisé, inclus dans un système d'information organisationnel.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Définitions :

Un projet informatique

- Un projet informatique est un projet dont les livrables sont des outils ou services informatiques (logiciels, systèmes d'information, sites web…). Il s'agit de projets généralement complexes. Ceci est principalement dû à la grande diversité des intervenants (techniciens, responsables métier, marketeurs, gestionnaires….) ainsi qu'à la difficulté de définir toutes les exigences.

- Le processus de développement d'un projet informatique passe par 5 phases :
  1. **Elaboration du schéma directeur** : Il s'agit d'une étude globale du système d'information à construire. Le but de cette étape est de réaliser le schéma directeur ainsi que Le plan de développement informatique
  2. **Etude préalable** : Il s'agit d'une étude critique de l'existant et de la définition des objectifs du nouveau système. Le but de cette étape est de produire un dossier d'étude et la prise de décision du choix de la solution.
  3. **Etude détaillée** : Il s'agit de fournir avec précision la description de la solution souhaitée : Définir logiquement les données et les traitements informatiques, les interfaces, le matériel… et construire le planning de réalisation. Le but de cette étape est de produire un cahier des charges fonctionnel et technique.
  4. **Réalisation** : Elle consiste à la production du logiciel, l'implantation des bases de données et la mise en place de la solution.
  5. **Mise en œuvre de la solution et assurer la maintenance** : Adapter la solution aux évolutions de l'environnement.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Définitions :

Le cahier des charges

- Le cahier des charges est un document essentiel à l'élaboration et la réalisation d'un projet. Il s'agit du document sur lequel les développeurs se basent pour concevoir et implémenter une base de données.
- Il présente une description détaillée du besoin des utilisateurs à savoir :
  - ✓ Le contexte général.
  - ✓ L'objectif du projet.
  - ✓ Les fonctionnalités attendues.
  - ✓ Les flux d'information et les processus métier.
  - ✓ Les règles de gestions des données.

- Il existe deux types de cahier des charges:
  - ✓ Le cahier des charges technique (CDCT): Il contient les exigences et contraintes techniques, économiques, industrielles, environnementales et matérielles d'un projet. Il sert à définir l'environnement technique: Architecture technique, les outils à utiliser, les technologies..
  - ✓ Le cahier des charges fonctionnel (CDCF) décrit la structure, les besoins et fonctionnalités attendues du maître d'ouvrage. Il contient les informations qui permettent d'addresser les exigences liées au projet en précisant les conditions de réalisation. Le CDCF doit comporter assez de détails pour être compréhensible par tous les acteurs du projet.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

La structure d'un cahier des charges

Un cahier de charge se compose de cinq elements essentiels :

1- Contexte et présentation du projet : On commence par présenter l'entreprise et l'importance du projet dans son plan stratégique. On définit aussi les acteurs cibles et les objectifs, et le périmètre du projet. Cette partie contient aussi la description de l'existant (si d'autres implémentations existent déjà)

Exemples :

Présenter l'entreprise:

- Le groupe Hospitalier SantéPro se compose de 4 hopitaux. Sa mission est de fournir des services de sante pour les habitants de la régions.

Présenter le projet:

- Refonte d'un système d'information hospitalier dans le but de :
- Augmenter la productivité du personnel
- Collecter plus d'information depuis les différents processus
- Minimiser le délais d'attente des patients

Définir le périmètre:

- La plateforme est utilisée par les différents hôpitaux du groupe, répartis sur la région. Il s'agit de plus de 2000 utilisateurs qui accèdent de manière journalière.

Copyright - Tout droit réservé - OFPPT

La structure d'un cahier de charge : (suite)

2- Description graphique et ergonomique : On y décrit la charte graphique ainsi que tous les éléments graphiques et ergonomiques exigés relatifs au nouveau projet.

Exemples :

- ✓ le logo
- ✓ la typographie
- ✓ les couleurs
- ✓ les illustrations

3- Description fonctionnelle et technique : Cette étape décrit qu'il faut définir toutes les spécifications techniques et fonctionnelles des livrables.

Exemples:

- ✓ Plateforme technique
- ✓ Technologies de développent
- ✓ Sécurité
- ✓ Données a collecter
- ✓ Règles de gestion

Copyright - Tout droit réservé - OFPPT

La structure d'un cahier de charge : (suite)

4- Définition des résultats attendus : On présente dans ce stade toutes les prestations attendues à la fin du projet ainsi que les délais de livraison:

Exemples:

- ✓ Livrer des exécutables /packages.
- ✓ Serveur web configuré et installé sur les lieux

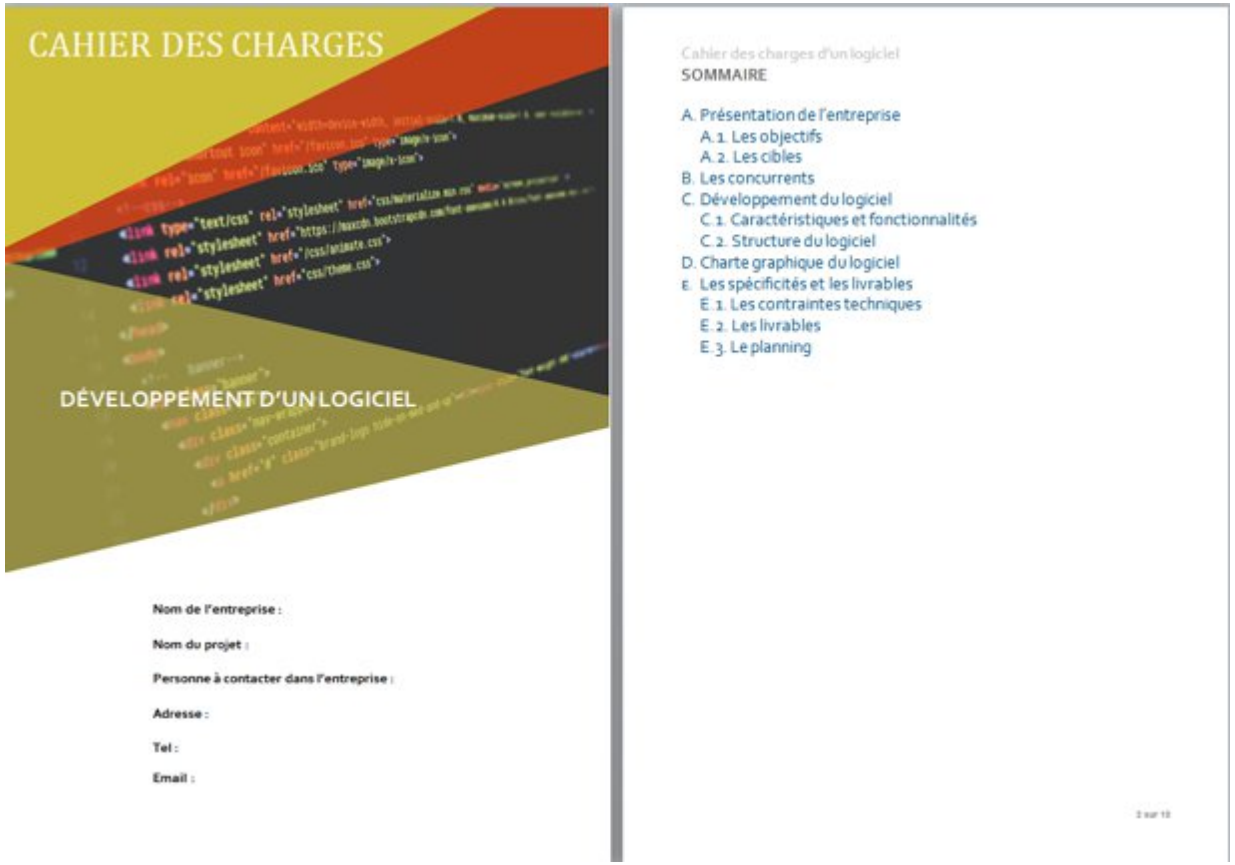
5- Budgétisation et fixation des délais : Cette phase concerne l'estimation du budget global permettant d'aiguiller les potentiels prestataires pour la réalisation de leurs devis :

- ✓ Un délais de réalisation de 200 jrs ouvrables
- ✓ Budget global de 1M de Dirhams.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

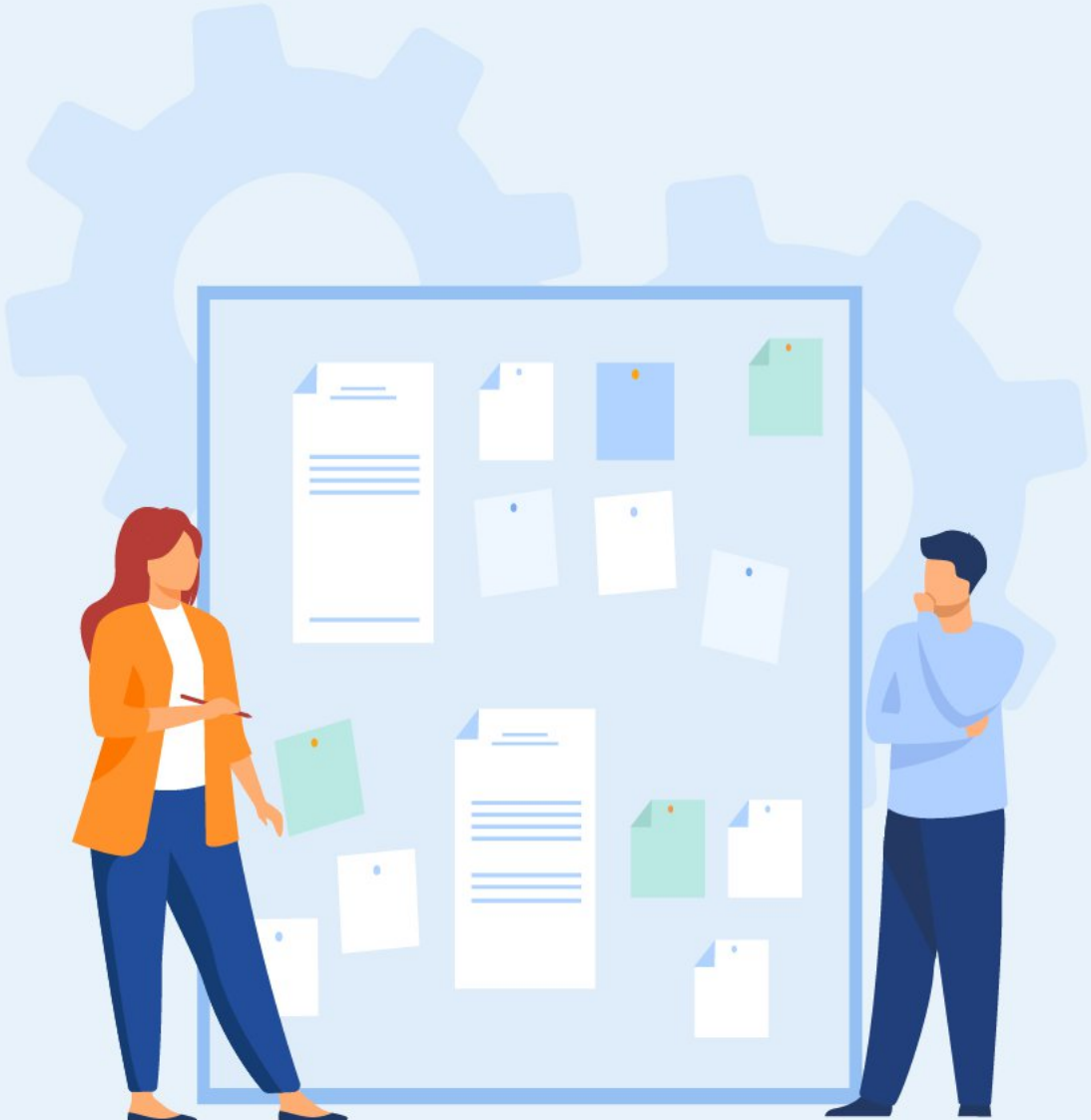
Exemple d'un cahier des charges



L'exemple ci-contre illustre les informations que l'on peut indiquer sur la page de couverture ainsi que les rubriques qui y figurent sur le sommaire.

Exemple de cahier des charges pour développement logiciel

Copyright - Tout droit réservé - OFPPT



CHAPITRE 1

ANALYSE DU CAHIER DES CHARGES

1. Lecture d’un cahier des charges
2. **Description des limites du projet**
3. Analyse des données et des traitements de la situation présentée

01 - ANALYSE DU CAHIER DES CHARGES

Description des limites du projet

PARTIE 1

Introduction :

- Le cahier des charges représente les attentes et les besoins du client ainsi que les contraintes du client.
- **En procédant à la lecture du cahier des charges, il faut définir le périmètre du projet :**
  - Le contexte du projet.
  - L'ensemble des données que le système est supposé gérer et stocker.
  - Les conditions et règles de gestion exprimées par le client.



Il faut aussi définir les limites du projet et répondre aux questions suivantes :

Quels sont les éléments de données cités par le document ?

Qui fait quoi ?

Quelles sont les données qui peuvent/doivent être précisées ?

Quand s'arrêter ? A-t-on tout pris en compte ?

Quelles limitations présente la situation actuelle ?

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Périmètre d'un projet :

Le périmètre du projet correspond à la délimitation précise du projet. Il s'agit de la liste des objectifs, des produits livrables, des affectations, des dépenses et des délais qui doivent être respectés. Ces termes de références définissent aussi les limites du projet. Si le périmètre d'un projet est efficacement tracé, la gestion des améliorations qui surviennent lors de la mise en œuvre et de la maintenance devient plus simple.

Concernant un projet lié aux Systèmes d'Information (mise en place d'un nouvel ERP, évolution d'un SI en fonction d'une nouvelle organisation, développement d'une plateforme web ...), le périmètre total est l'identification et le recensement des applications/modules impactés par le projet.

A partir du cahier des charges ainsi que des échanges avec les porteurs du projet, on peut définir le périmètre et dresser les limites du projet en suivant les étapes suivantes :

1. Définir les buts : Il s'agit des objectifs à réaliser par le biais du projet.

Exemple : Dans le cas d'un projet d'informatisation des activités d'un centre de formation, le but serait de : *créer des formulaires d'inscriptions pour les étudiants qui vont simplifier les processus d'inscription et leur prise en charge et de suite assurer l'accès aux informations nécessaires a la bonne gestion du centre.*

2. Définir les livrables : Il faut identifier les résultats attendus du projet : c'est-à-dire l'ensemble des livrables. L'identification des livrables permet de détecter les dérives des objectifs si celles-ci surviennent.

Exemple : Quelles choses tangibles nous devons créer pour le compte du client (Centre de formation) ? Dans ce cas, il s'agit du formulaire informatise ainsi que la base de données des inscriptions.

Copyright - Tout droit réservé - OFPPT

3. Définir les tâches et les activités du projet

Il s'agit des moyens qui vont permettre la création des livrables et la réalisation des but du projet. Les livrables sont ainsi découpés en tâches et activités distinctes. Ceci permet de faciliter la gestion des projets surtout quand la complexité est importante.

Exemple : Pour créer le formulaire d'inscription on doit :

- Choisir et rédiger un exemplaire du formulaire
- Développer une interface de saisie du formulaire.
- Créer une base de donner pour stocker les informations du formulaire…

4. Définir les contraintes du projet

Les trois principales contraintes d'un projet sont le budget, le temps et la portée:

Contrainte de budget ou cout : L'ensemble des ressources financières (frais du matériels, services et ressources humaines) nécessaires pour la réalisation du projet dans le respect les limites et délais prédéfinis.

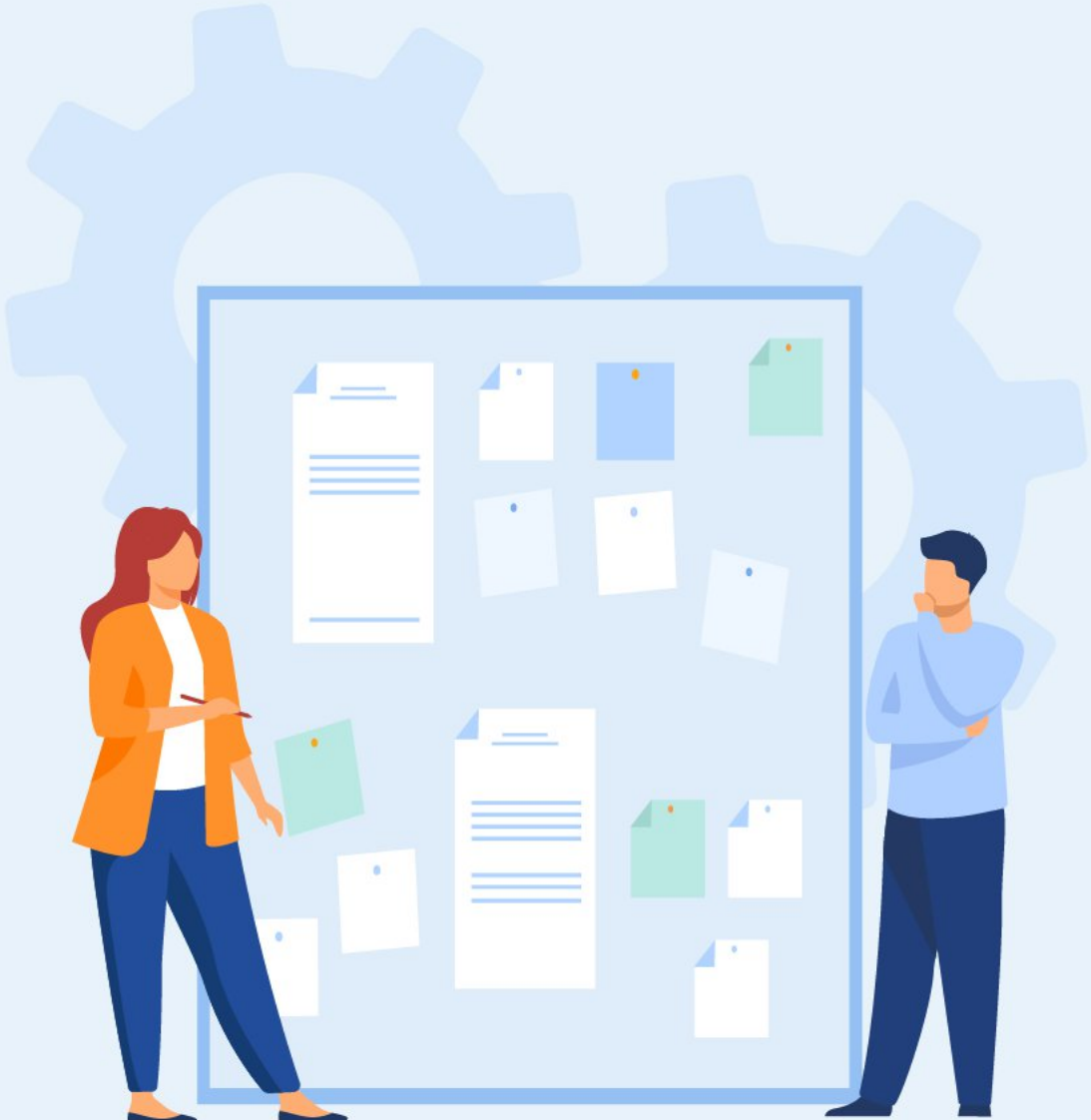
Contrainte de temps : Le calendrier de livraison du projet en totalité ainsi que des différentes phases du projet.

Contrainte de portée : La définition des objectifs, des livrables, des fonctionnalités et des tâches à accomplir pour la finalisation du projet.

***Exemple* :** Voici quelques contraintes applicables à votre questionnaire client :

-Le projet doit être bouclé en 6 mois
- Le budget total pour le projet ne doit pas dépasser 50 000 Dirhams
- L'équipe de développement ne pourra pas finaliser la conception dans 3 mois.

Copyright - Tout droit réservé - OFPPT



CHAPITRE 1

ANALYSE DU CAHIER DES CHARGES

1. Lecture d’un cahier des charges
2. Description des limites du projet
3. **Analyse des données et des traitements de la situation présentée**

01 - ANALYSE DU CAHIER DES CHARGES

Analyse des données et des traitements de la situation présentée

PARTIE 1

Introduction :

- On inclut souvent dans un projet informatique : les bases de données et le système informatique comprenant les ressources et infrastructures réseau, les applications et aussi les règles et dispositifs de sécurité.
- L'élaboration des bases de données afin de gérer l'accès aux données, le stockage et le traitement représentent un pilier du livrable d'un projet.
- Le cahier des charges relatif à un projet informatique indique les différents volets qui concernent ce projet, notamment la gestion des données.

Comment, à partir de ce cahier de charges, élaborer la solution souhaitée ?

- Il faut utiliser des méthodes de modélisation et de conception du système et de la base de données.
- Dans ce qui suit, on s'intéresse au volet du cahier des charges qui concerne la base de données.

Copyright - Tout droit réservé - OFPPT

Définition :

- Une des fonctions d'un système informatique dans une organisation est de stocker et gérer les données nécessaires à son bon fonctionnement, d'où la nécessité du concept des bases de données.

Une base de données

- **Une base de données** est une structure permettant de stocker un grand nombre d'informations afin d'en faciliter l'utilisation.

Objectifs :

- Centraliser le stockage des informations
- Faciliter l'accès à l'information
- Assurer la justesse et la cohérence des informations stockées surtout lors des modifications
- Garantir l'intégrité et la confidentialité des données

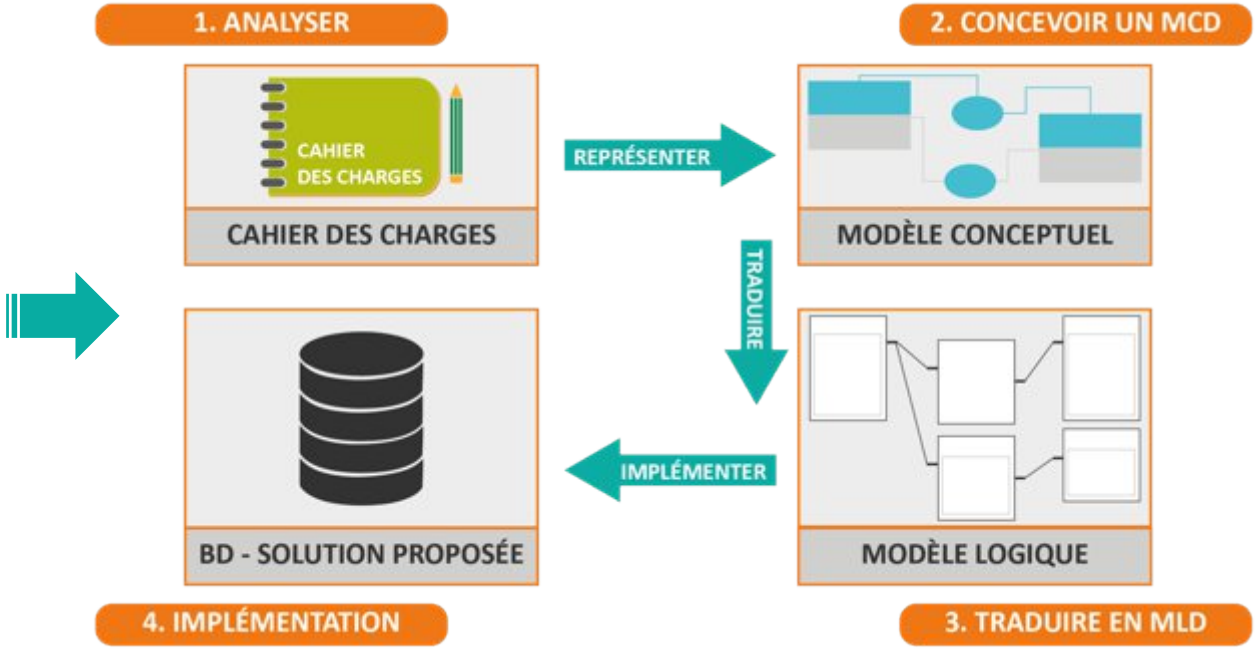
Copyright - Tout droit réservé - OFPPT

01 - ANALYSE DU CAHIER DES CHARGES

Analyse des données et des traitements de la situation présentée

- La conception d'une base de données passe par quatre phases comme illustré par le schéma :
    1. **Analyse du cahier des charges et clarification du besoin du client.**
    2. **Conception d'un modèle conceptuel qui représente tous les éléments nécessaires du projet.**
    3. **Traduction du modèle conceptuel en modèle logique.**
    4. **Implémentation de la base de données proposée.**

Dans le schéma ci-contre, la première étape de la conception d'une base de données se base sur l'analyse pertinente du cahier des charges et la bonne compréhension des besoins exprimés par les utilisateurs. Elle est par la suite essentielle et délicate en même temps.



Les phases de conception d'une base de données

PARTIE 1

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Exemple : Cahier des charges du projet: Gestion d'un centre de formation

- Un centre de formation désire stocker et gérer des données concernant les étudiants et les formations dans lesquelles ils sont inscrits. Le travail demandé est la modélisation des données persistantes et la représentation sous forme tabulaire de ces données telles qu'elles seront stockées dans la base de données.

- Les étudiants choisissent la formation et la session de cette formation dans laquelle ils veulent s'inscrire et payent le prix de la formation.

- Un étudiant est définit par **son numéro de CIN**. Il est, lors de son inscription, amené à remplir une fiche contenant son nom et prénom, sa date de naissance, son adresse, sa ville et son niveau scolaire.
- Puis, depuis le catalogue des **formations**, il doit choisir la formation souhaitée, et la **session** relative à cette formation. Il indique aussi le **type de cours** qu'il veut suivre (présentiel ou à distance). Une fiche d'inscription est conservée par l'administration (voir annexe).

Définition de l'**objectif** de la base de données :
- Gérer les données des étudiants, formations et inscriptions.

Définition des **processus** métier :
- Gérer les données des étudiants, formations et inscriptions.

Définition des **données** de la base de données :
- Étudiants (CIN, nom, prénom…).
- Formations, sessions, type des cours…

Il faut consulter les documents en annexe pour voir les fiches actuelles contenant la liste des données collectées.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Exemple : (suite)

- Pour chaque formation, le catalogue précise le code, le titre, la durée, le prix et les spécialités (code et nom) qui concernent cette formation ainsi que les sessions ouvertes avec leurs date début et date fin.

Définition des **données** de la base de données :

- Formation (code, titre, durée, prix…)
- Spécialité…
- Session (date début, date fin…)

- Voici quelques règles de gestion mises en œuvre par la direction du centre :
  - Un étudiant peut être inscrit dans plusieurs sessions de formations.
  - La formation peut se tenir en plusieurs sessions.
  - Un étudiant ne peut pas être inscrit à plusieurs sessions de la même formation.
  - Une formation n'est ouverte que s'il y a plus de 10 étudiants inscrits.
  - Une formation peut faire partie de plusieurs spécialités.

Définition **des règles de gestion** :

- Conditions et contraintes à respecter lors de la modélisation de la base de données.

Copyright - Tout droit réservé - OFPPT

- Les règles de gestion ainsi que les informations collectées permettent de définir les éléments de la base de données que nous allons construire, les relations entre ces éléments et aussi d'assurer **l'intégrité** des données :
    - Exhaustivité
    - Exactitude
    - Cohérence des données

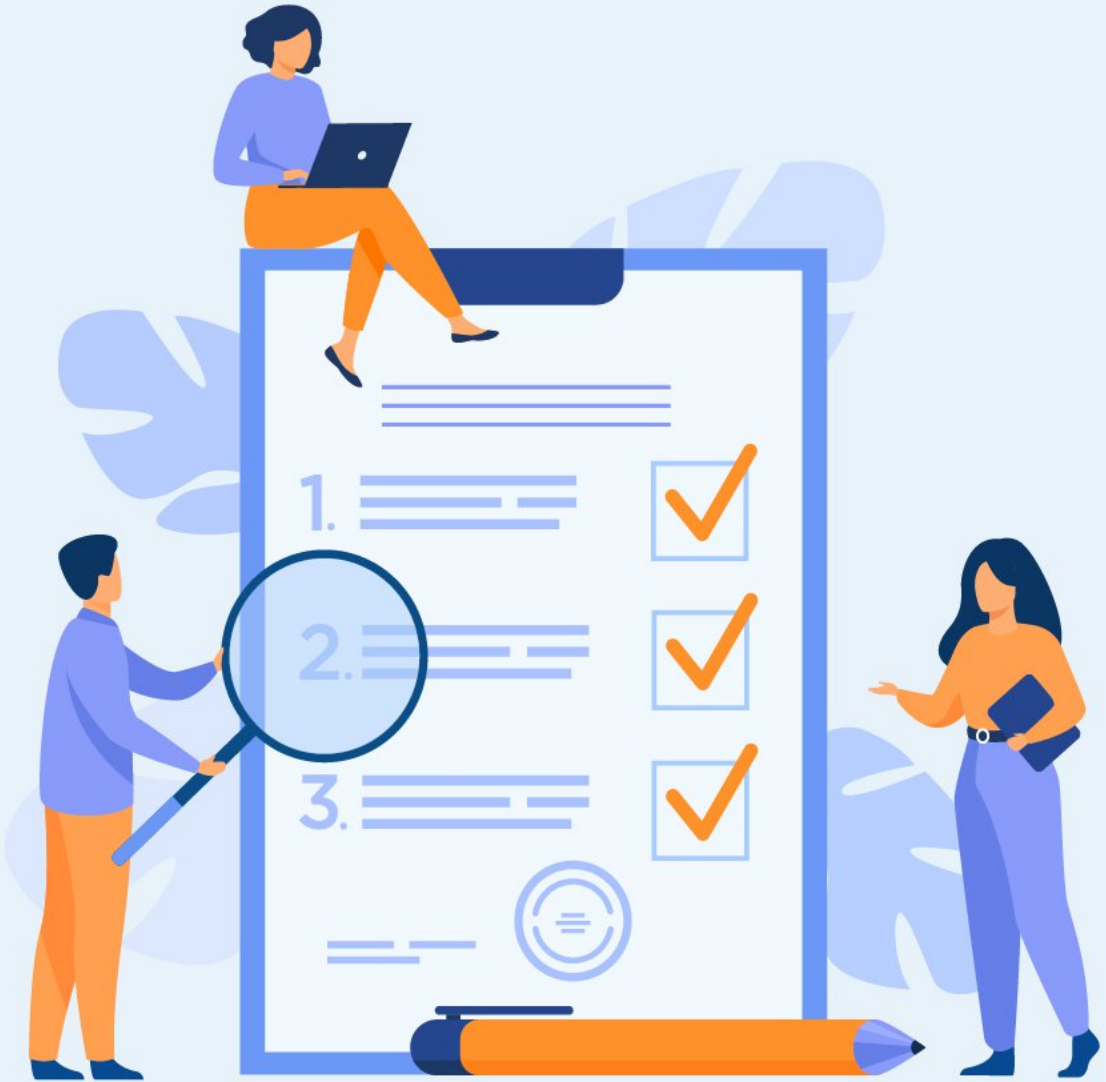
Exemple :

- Si l'on veut modéliser les données de ce centre, nous allons créer un ensemble de tables liées entre elles par des relations :
    - Une table (ou entité) **ÉTUDIANT** qui contiendra des attributs :
        - Nom
        - Prénom
        - Adresse…
    - Une table (ou entité) **FORMATION** qui contiendra des attributs :
        - Titre
        - Durée
        - Prix…
    - Une table (ou entité) **SESSION** qui contiendra des attributs :
        - Code Formation
        - Date
        - Lieu…

Copyright - Tout droit réservé - OFPPT

CHAPITRE 2

MODÉLISATION DES DONNÉES



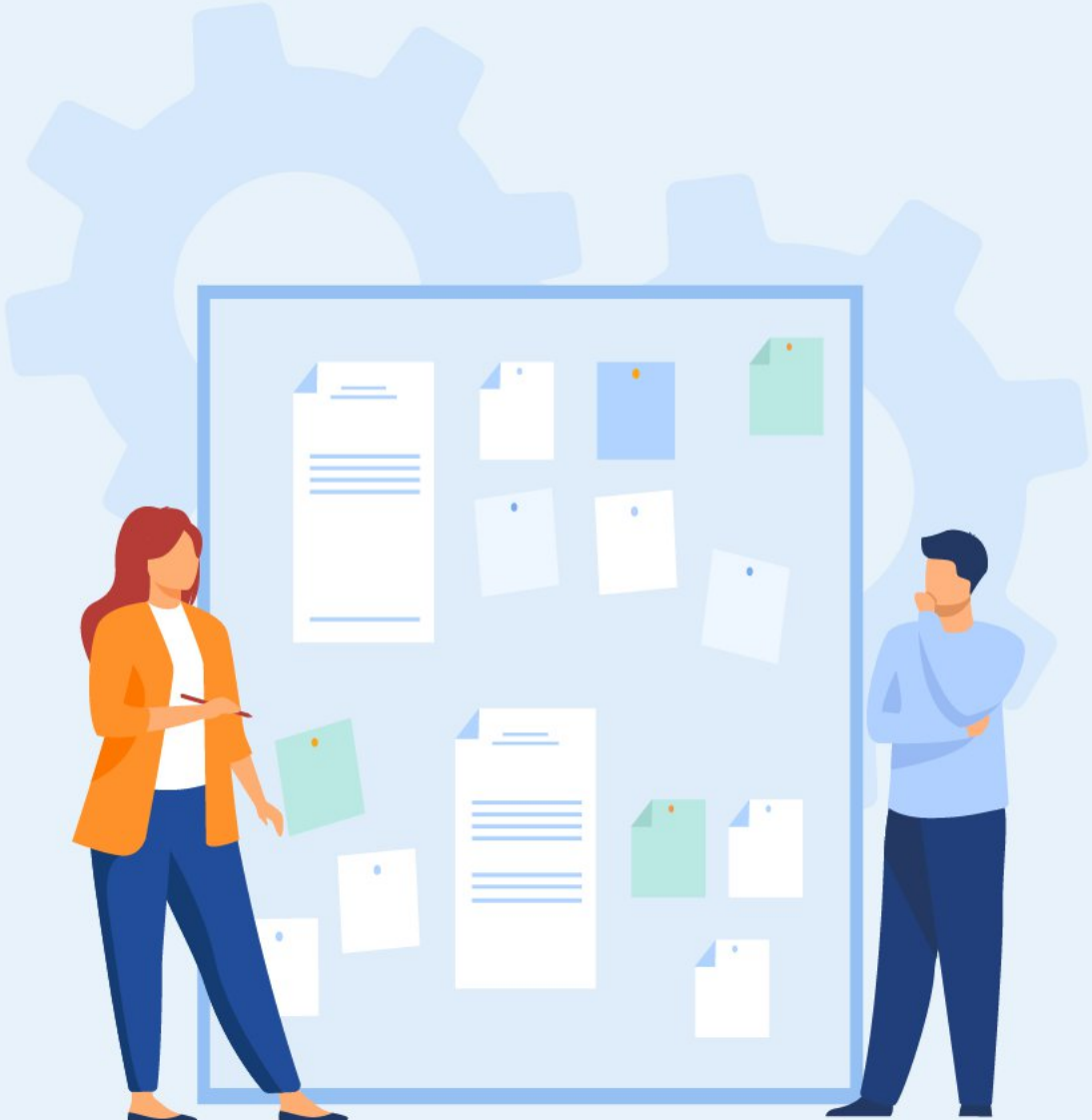
Ce que vous allez apprendre dans ce chapitre :

- L'élaboration des dictionnaires de données
- L'identification des dépendances fonctionnelles
- La construction du Modèle Conceptuel de Données (MCD)

08 heures

CHAPITRE 2

MODÉLISATION DES DONNÉES



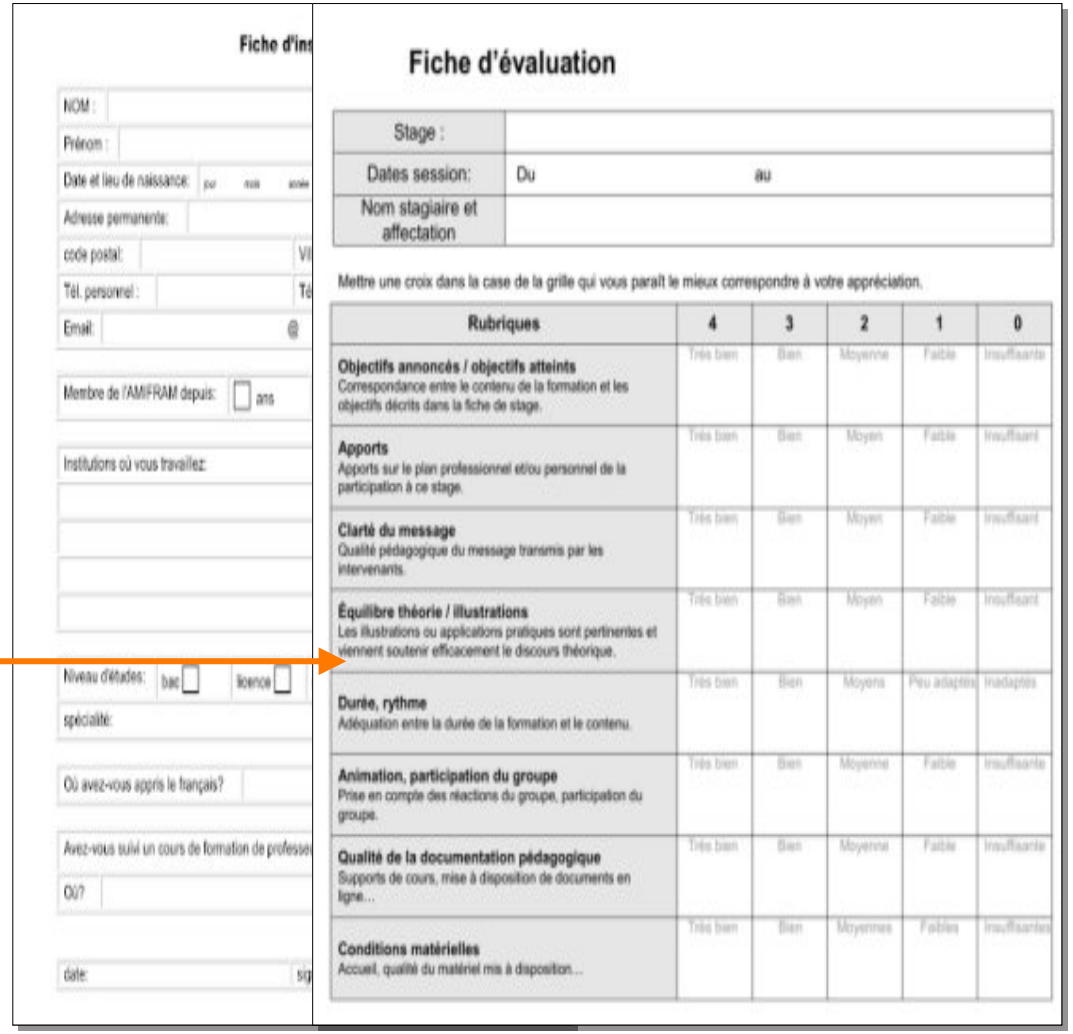
1. **Contraintes déduites des règles de gestion**
2. Dictionnaire de données
3. Construction du graphe de dépendances fonctionnelles
4. Règles de passage du graphe au modèle conceptuel de données
5. Construction du modèle conceptuel de données

02 - Modélisation des données

Contraintes déduites des règles de gestion

- Les règles de gestion fournies par le cahier des charges permettent d'identifier les éléments de données de la base à concevoir. Ces règles doivent être traduites en contraintes afin d'assurer l'intégrité des données et la validation des modèles à construire.
- **Identification des éléments de données :** Exemple du cahier des charge du centre de formation :

Les attributs des entités **«ÉTUDIANT», « FORMATION », « SESSION » et « SPÉCIALITÉ »** peuvent être déduits du texte ainsi que des fiches de renseignement données en annexe du cahier des charges.



Fiche d'évaluation

Stage :	
Dates session:	Du au
Nom stagiaire et affectation	

Mettre une croix dans la case de la grille qui vous paraît le mieux correspondre à votre appréciation.

Rubriques	4	3	2	1	0
Objectifs annoncés / objectifs atteints Correspondance entre le contenu de la formation et les objectifs décrits dans la fiche de stage.	Très bien	Bien	Moyenne	Faible	Insuffisante
Apports Apports sur le plan professionnel et/ou personnel de la participation à ce stage.	Très bien	Bien	Moyen	Faible	Insuffisant
Clarté du message Qualité pédagogique du message transmis par les intervenants.	Très bien	Bien	Moyen	Faible	Insuffisant
Équilibre théorie / illustrations Les illustrations ou applications pratiques sont pertinentes et viennent soutenir efficacement le discours théorique.	Très bien	Bien	Moyen	Faible	Insuffisant
Durée, rythme Adéquation entre la durée de la formation et le contenu.	Très bien	Bien	Moyens	Peu adaptés	Inadaptés
Animation, participation du groupe Prise en compte des réactions du groupe, participation du groupe.	Très bien	Bien	Moyenne	Faible	Insuffisante
Qualité de la documentation pédagogique Supports de cours, mise à disposition de documents en ligne…	Très bien	Bien	Moyenne	Faible	Insuffisante
Conditions matérielles Accueil, qualité du matériel mis à disposition…	Très bien	Bien	Moyennes	Faibles	Insuffisantes



PARTIE 1

Copyright - Tout droit réservé - OFPPT

- Toujours dans le même sillage de l'exemple du cahier des charges relatif au centre de formation, les contraintes ci-après ont été identifiées :
  - Un étudiant peut être inscrit dans plusieurs sessions de formations.
  - La formation peut se tenir en plusieurs sessions.
  - Un étudiant ne peut pas être inscrit à plusieurs sessions de la même formation.
  - Une formation n'est ouverte que s'il y a plus de 10 étudiants inscrits.
  - Une formation peut faire partie de plusieurs spécialités.

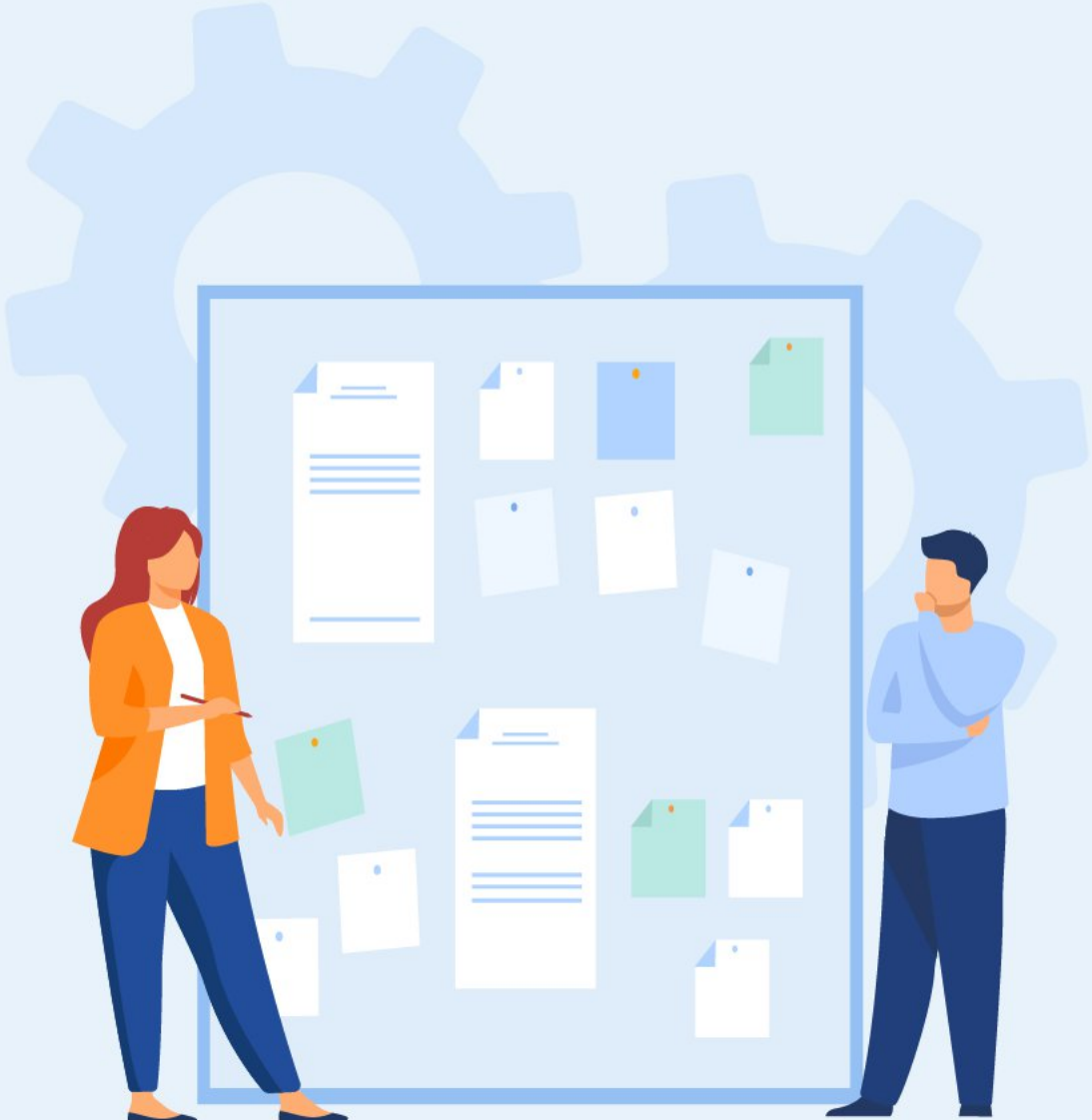
- Le tableau ci-après récapitule les règles de gestion relatives toujours au même exemple :

Règle N°	Énoncé de la règle
1	Un élément de l'entité ÉTUDIANT peut être associé à plusieurs éléments de l'entité SESSION.
2	Un élément de l'entité SESSION concerne un élément unique de l'entité FORMATION.
3	Un élément de l'entité FORMATION peut être associé à plusieurs éléments de l'entité SESSION.
4	Un élément de l'entité FORMATION peut être associé à un ou plusieurs éléments de l'entité SPÉCIALITÉ.

Copyright - Tout droit réservé - OFPPT

CHAPITRE 2

MODÉLISATION DES DONNÉES



1. Contraintes déduites des règles de gestion
2. **Dictionnaire de données**
3. Construction du graphe de dépendances fonctionnelles
4. Règles de passage du graphe au modèle conceptuel de données
5. Construction du modèle conceptuel de données

PARTIE 1

Avant la phase de conception du modèle conceptuel des données, il faut **relever** d'abord trois types d'informations à partir du cahier des charges.

Concept :
Il s'agit d' « objets » ou choses qui vont par la suite être des entités du schéma entité-association. Ce sont des éléments complexes qui peuvent être décomposés en plusieurs informations sous formes de « données ».

Exemples : étudiant, formation, session…
Contre-exemple : CIN, durée de la formation, nom de la session…

Donnée :
C'est une information élémentaire, qui ne peut pas être décomposée. Elle se trouve souvent liée à un concept.

Exemples : CIN, durée de la formation, nom de la session, salle, adresse étudiant…
Contre-exemple : numéro (sans préciser relatif à qui ou à quoi)..

Valeur :
Il s'agit d'occurrences ou exemples des données d'un concept.

Exemples : «G434568» valeur de la donnée : numéro de la CIN, relative au concept : étudiant.
Contre-exemples : nom de la formation, numéro de la CIN, étudiant…

Copyright - Tout droit réservé - OFPPT

Afin de réaliser un bon relevé d'informations, il faut :

- Bien identifier un **concept** : identifier les noms des objets, choses, personnes et types qui ont des données y afférentes.
- Relever uniquement les concepts et données qui concernent le système à concevoir. (faire attention aux détails inutiles).

Exemple :

- Le centre de formation « CF excellence » offre des formations aux étudiants des villes suivantes : Tanger, Rabat, Casablanca.
- **On constate que :**
    - « CF excellence » n'est pas une donnée, c'est le nom de l'organisation.
    - Formation, étudiant, session, spécialité sont des concepts.
    - Il y a une donnée rattachée au concept Étudiant qui est « Nom Ville ».
    - Tanger, Rabat, Casablanca sont des valeurs de la donnée « Nom Ville ».
    - Il y a un lien entre les concepts Formation et Étudiant.
    - …

Copyright - Tout droit réservé - OFPPT

Dictionnaire des données : définitions

- Le dictionnaire des données contient toutes les données nécessaires qui vont être conservées dans la base de données. Il est souvent présenté sous forme d'un tableau qui indique pour chaque donnée les informations suivantes :
  - **Le code** : il s'agit d'un libellé désignant une donnée.
  - **La désignation** : description de la donnée.
  - **Le type de données** :
    - **Alphabétique** : lorsque les valeurs de la donnée sont composées de caractères alphabétiques.
    - **Numérique** : lorsque les valeurs de la donnée sont composées de nombres.
    - **Alphanumérique** : lorsque les valeurs de la donnée sont composées de caractères alphabétiques et numériques.
    - **Date** : quand il s'agite d'une date.
    - **Booléen** : vrai ou faux.
  - **La taille** : elle exprime la longueur des valeurs.
  - **Observations** : qui peut contenir des informations complémentaires.

Copyright - Tout droit réservé - OFPPT

Dictionnaire des données : définitions

- Après le relevé des données, il faut préciser :

1 - LA NATURE DE CHAQUE DONNÉE

Est-ce qu'il s'agit d'un concept, d'une donnée ou d'une valeur ?

2 - SON RÔLE

Est-ce qu'elle identifie un concept, est-ce qu'elle est calculée ou élémentaire ?

3 - LE CONCEPT AUQUEL ELLE APPARTIENT

Est-ce qu'elle appartient à un ou plusieurs concepts ?

Copyright - Tout droit réservé - OFPPT

- Ensuite, procéder aux contrôles suivants :
  - **Imprécisions :** s'assurer que les champs sont bien nommés et décrits

    Exemple : « Ville » tout court n'est pas précis.

    ➢ On optera plutôt pour « Ville étudiant ».
  - **Polysémies :** il s'agit de deux données portant le même nom mais qui désignent des choses différentes.

    Exemple : « Nom » relatif a l'étudiant et « Nom » relatif à la session.

    ➢ Il faut renommer les deux données en : Nom étudiant et Nom Session.
  - **Synonymes :** il s'agit de deux descriptions différentes qui désignent la même donnée.

    Exemple : code CIN et numéro CIN.

    ➢ Il faut garder une seule description et supprimer les autres synonymes.
  - **Les données calculées** ne doivent pas figurer dans le dictionnaire des données, mais plutôt, il faut préciser les éléments qui ont permis ce calcul.

    Par exemple : PRIX TTC = PRIX HT * TVA

    ➢ Les données à retenir seront : PRIX HT et TVA.

Copyright - Tout droit réservé - OFPPT

Bonne pratique :

- Afin d'uniformiser la nomenclature des données, on adopte une formule se composant de : objet (en minuscules) + raccourcis de nom du concept qu'elle représente (première lettre en majuscule).

Exemple :

- Numéro CIN de l'étudiant ☾ numCINEtu
- Titre de la formation ☾ titreForm

Le dictionnaire de données relatif à notre exemple de la gestion du centre de formation est comme suit : (1/2)

Code donnée	Désignation	Type	Taille	Observation
numCINEtu	Numéro CIN	Alphanumérique	9	Identifiant de l'étudiant
nomEtu	Nom de l'étudiant	Alphabétique	30	
prenomEtu	Prénom de l'étudiant	Alphabétique	30	
dateNaissEtu	Date de naissance	Date		
niveauEtu	Niveau scolaire	Alphanumérique	15	
nomvilleEtu	Nom de la ville	Alphabétique	15	
AdresseEtu	Adresse de l'étudiant	Alphanumérique	90	

Copyright - Tout droit réservé - OFPPT

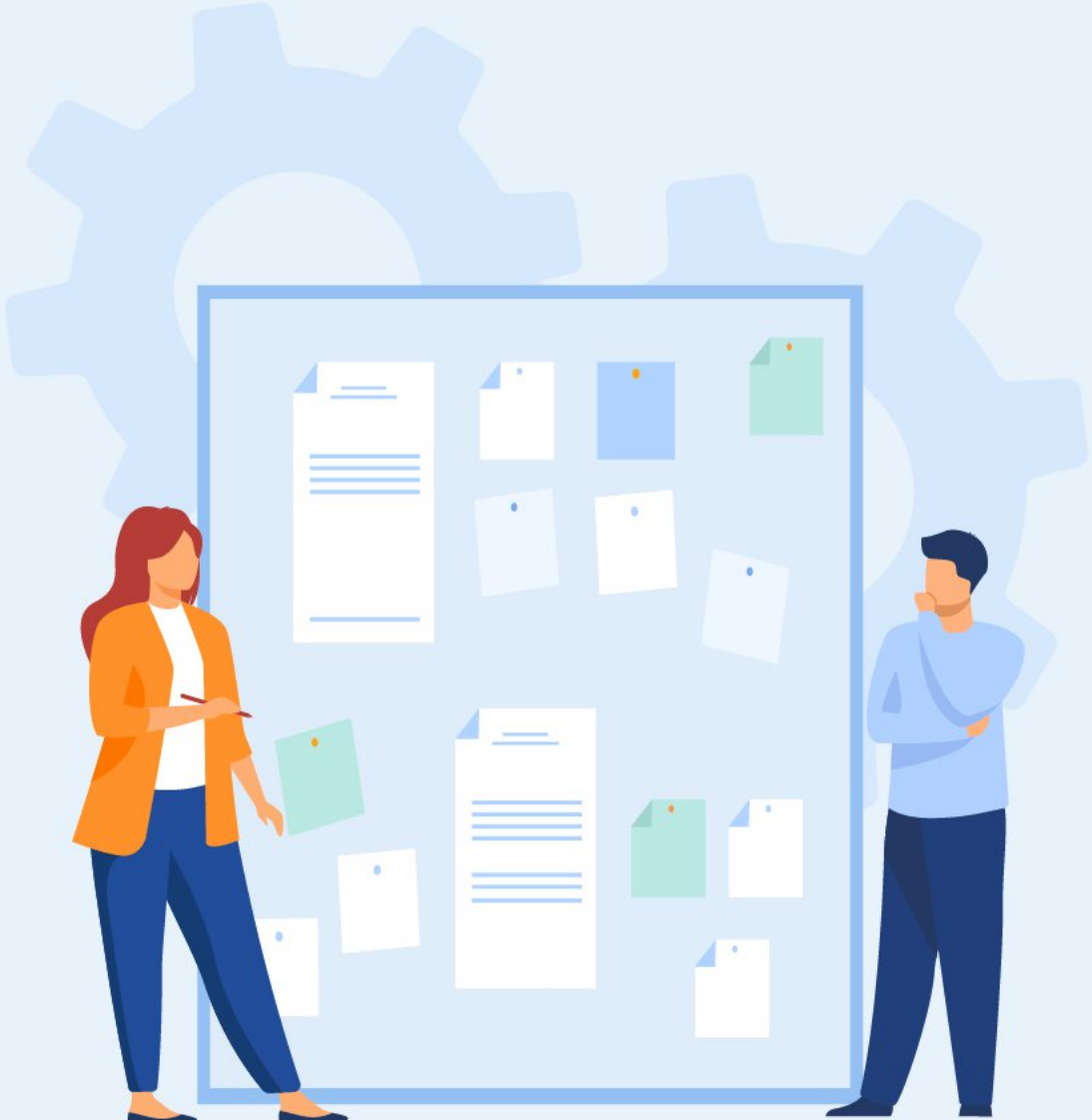
- **Le dictionnaire de données relatif à notre exemple de la gestion du centre de formation : (2/2)**

Code donnée	Désignation	Type	Taille	Observation
codeForm	Code de la formation	Alphanumérique	9	Identifiant de la formation
titreForm	Titre de la formation	Alphanumérique	30	
dureeForm	Durée de la formation	Numérique	3	
prixForm	Prix de la formation	Numérique	5	
codeSess	Code de la session	Alphanumérique	9	Identifiant de la session
nomSess	Nom de la session	Alphanumérique	30	
dateDebutSess	Date du début de la session	Date		
dateFinSess	Date de la fin de la session	Date		
codeSpec	Code de la spécialité	Alphanumérique	10	
nomSpec	Nom de la spécialité	Alphanumérique	30	
descSpec	Description de la spécialité	Alphanumérique	90	
typeCours	Type de cours	Alphabetique		Distanciel ou presentiel

Copyright - Tout droit réservé - OFPPT

CHAPITRE 2

MODÉLISATION DES DONNÉES

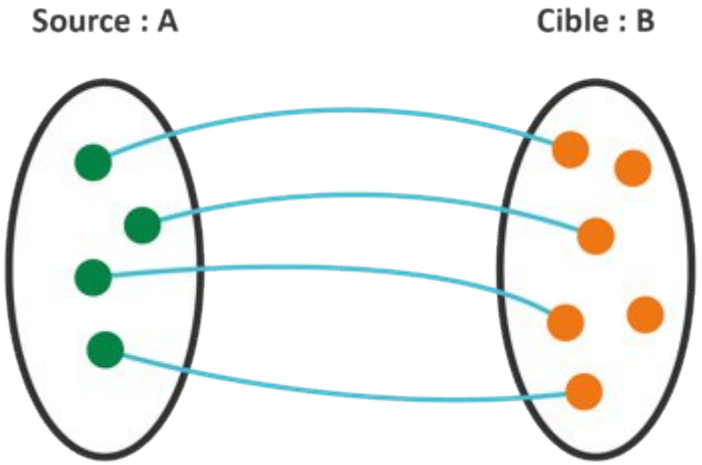


1. Contraintes déduites des règles de gestion
2. Dictionnaire de données
3. **Construction du graphe de dépendances fonctionnelles**
4. Règles de passage du graphe au modèle conceptuel de données
5. Construction du modèle conceptuel de données

- Après le recensement des données dans un dictionnaire de données, l'étape suivante est celle de découvrir les relations entre ces données.

Dépendances fonctionnelles : Définition

- Soit deux groupes de données : A (source) et B (cible).
- On entend par dépendance fonctionnelle une relation entre les deux groupes de données A et B de telle façon que :
- Un élément du groupe A (source) permet de déterminer un et un seul élément du groupe B (cible).



Dépendances entre les éléments des deux ensembles A et B

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Exemple :

Code de Formation	Titre de Formation	Durée	Prix
ID01	Introduction au développement	3 mois	2500
CCP01	C/C++	30 jours	3000
ID02	Introduction au développement	3 mois	2700
BD001	Base de données	30 jours	2500

- Pour le concept Formation, la valeur du code de formation ID01 détermine que le titre de la formation est « Introduction au développement ».
- On peut déduire que :
  - Le code de formation **détermine une seule occurrence** du titre de formation.
  - Le titre de formation **dépend** du code de formation.
  - Cette relation est symbolisée sous cette forme : codeForm -> titreForm.
  - Cette relation n'est pas réversible : plusieurs formations peuvent avoir le même titre.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

La liste des dépendances fonctionnelles :

- La liste des dépendances fonctionnelles est élaborée à partir du dictionnaire des données.
- Il ne faut retenir que les dépendances directes et donc éliminer les transitivités :
- C'est-à-dire que si D1 -> D2 et D2-> D3, alors D1->D3 est obtenue par transitivité et n'est pas, par la suite, directe.

Exemple :

- Voici la liste des dépendances fonctionnelles construite à partir du dictionnaire de données de notre exemple du centre de formation :

SOURCE	CIBLE
numCINEtu ☾	nomEtu
numCINEtu ☾	prenomEtu
numCINEtu ☾	dateNaissEtu
numCINEtu ☾	niveauEtu
numCINEtu ☾	nomvilleEtu
numCINEtu ☾	AdresseEtu

SOURCE	CIBLE
codeSpec ☾	nomSpec
codeSpec ☾	descSpec
codeSpec ☾	codeForm

SOURCE	CIBLE
codeSess ☾	nomSess
codeSess ☾	codeForm
codeSess ☾	dateDebutSess
codeSess ☾	dateFinSess

SOURCE	CIBLE
codeForm ☾	titreForm
codeForm ☾	dureeForm
codeForm ☾	prixForm

SOURCE	CIBLE
numCINEtu + codeSess ☾	typeCours

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Contre-exemples :

SOURCE	CIBLE	REMARQUE
numCINEtu ☾	nomForm	1) Un étudiant peut être inscrit dans plusieurs formation à la fois.
codeform ☾	nomSpec	2) Une formation peut concerner plusieurs spécialités.
NomEtu ☾	NomForm	3) NomEtu n'induit pas NomForm, aucune relation directe n'existe entre les deux.

1) Cette dépendance suggère que l'on peut déduire le nom de la formation à partir du code de CIN de l'étudiant. Ceci n'est pas correct du à la règle de gestion qui indique qu'un élève peut être inscrit dans plusieurs formations.

2) Cette dépendance suggère que l'on peut déduire le nom de la spécialité à partir du code de la formation. Ceci n'est pas correct du à la règle de gestion qui indique qu'une formation peut apparaitre dans le curriculum de plusieurs spécialités.

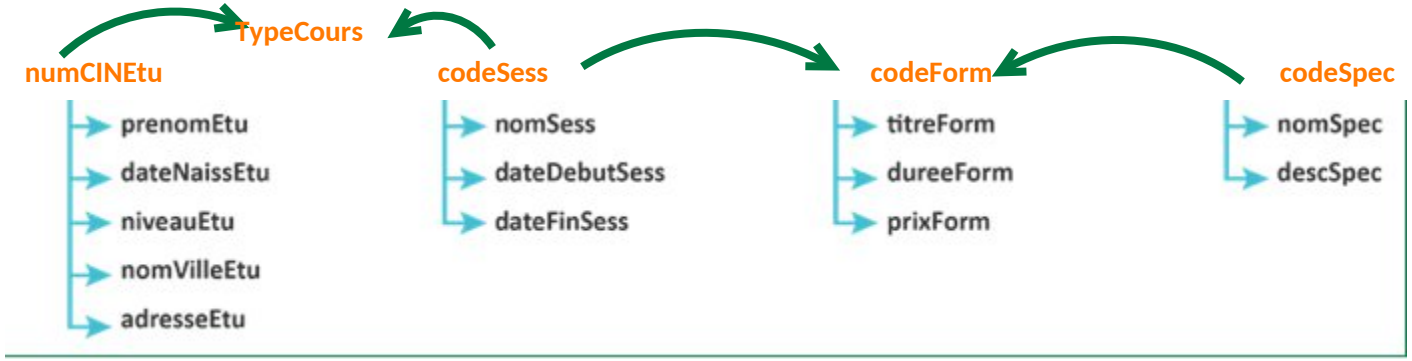
3) Cette dépendance suggère que l'on peut déduire le nom de la formation à partir du nom de l'étudiant. Ceci n'est pas correct car d'une part, le nom de l'étudiant n'est pas un identifiant, et qu'aucune relation directe n'existe entre ces deux attributs.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Le graphe des dépendances fonctionnelles :

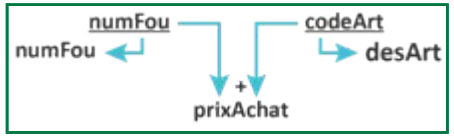
- Le graphe des dépendances fonctionnelles est la représentation graphique des dépendances fonctionnelles entre les données. Dans le cas du même exemple, le graphe des dépendances fonctionnelles est le suivant :



- On remarque que la donnée **typeCours** dépend de la combinaison de numCINEtu et codeSess.

Par exemple :

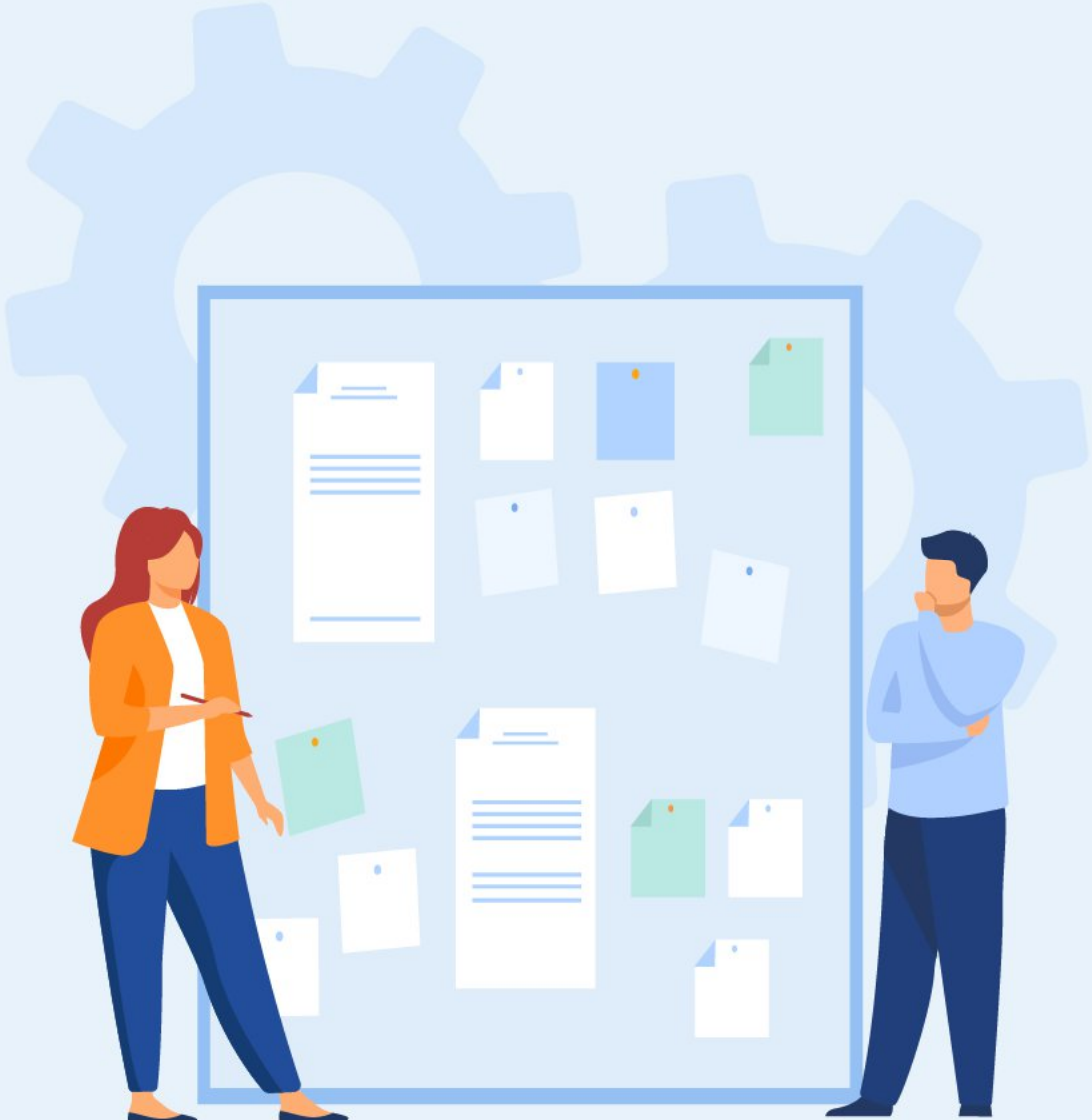
- Le prix d'un article varie d'un fournisseur à l'autre.
- La DF sera présentée ainsi : numFourisseur + CodeArticle -> PrixArticle
- Le signe + indique que les données numFourisseur et CodeArticle doivent être groupées pour obtenir le prix de l'article.



Copyright - Tout droit réservé - OFPPT

CHAPITRE 2

MODÉLISATION DES DONNÉES



1. Contraintes déduites des règles de gestion
2. Dictionnaire de données
3. Construction du graphe de dépendances fonctionnelles
4. **Règles de passage du graphe au modèle conceptuel de données**
5. Construction du modèle conceptuel de données

PARTIE 1

- La phase de conception des systèmes d'information, et bien évidemment les bases de données, nécessite le recours à des méthodes de modélisation. C'est-à-dire la représentation virtuelle des processus et données de telle façon à bien comprendre l'existant et bien définir les futures livrables.

  Il existe plusieurs méthodes d'analyse et de conception, une des méthodes les plus utilisées étant la méthode **MERISE**.

Qu'est-ce que la méthode Merise ?

- **La méthode Merise (Méthode d'étude et de réalisation informatique pour les systèmes d'entreprise)** date de la fin des années 1970 en France. Il s'agit d'une méthode d'analyse et de conception de systèmes d'information qui se base sur le principe de la séparation des données et des traitements.
- La méthode Merise propose une démarche basée sur trois niveaux (ou cycles) : la conception, l'organisation et la technique. En effet, modéliser un système revient à produire une analyse globale de sa fonction : Décrire ce qu'il fait avant de se focaliser sur comment il le fait. Les données étant séparées des traitements, il faut vérifier la concordance entre données et traitements afin de vérifier que toutes les données nécessaires aux traitements sont présentes et qu'il n'y a pas de données superflues.Les trois niveaux de représentation des données, sont détaillés ci-dessous.

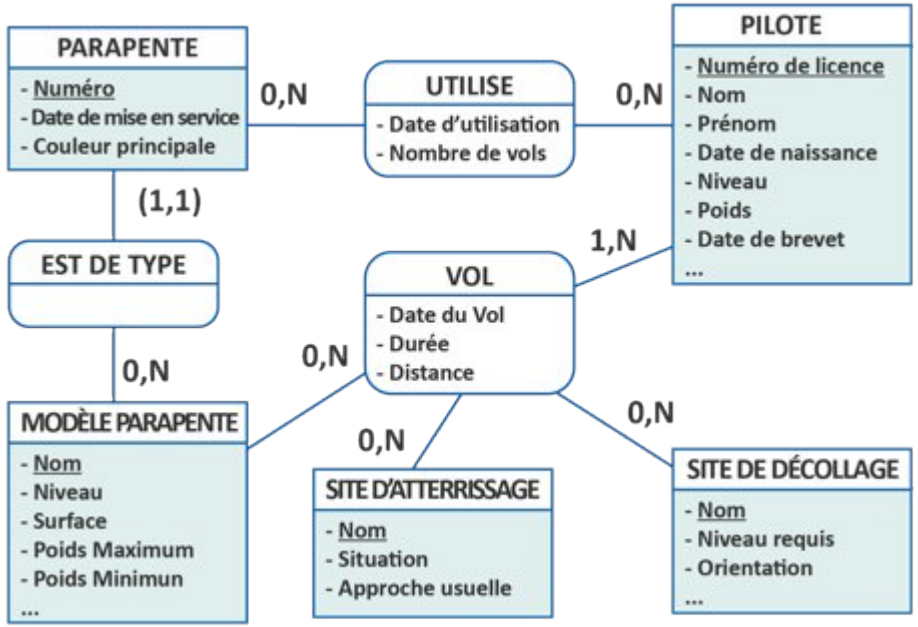
- **Niveau conceptuel :** le *modèle conceptuel des données (MCD)* décrit les entités du monde réel, en terme d'objets, de propriétés et de relations, indépendamment de toute technique d'organisation et d'implantation des données.
- **Niveau logique :** le *modèle logique des données (MLD)* adapte le modèle conceptuel au contexte organisationnel. Il s'agit d'une transcription du MCD dans un formalisme adapté à une implémentation ultérieure sous forme de base de données.
- **Niveau physique :** le *modèle physique des données (MPD)* permet d'établir la manière concrète dont la base de données sera construite.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Le modèle conceptuel de données :

- Le modèle conceptuel des données (MCD) formalise les données qui vont être stockées dans la base de données.
- Il s'agit donc d'une représentation des données, facile à comprendre, et qui permet de décrire la base de données à l'aide d'entités. La description par la méthode des entités association (MERISE) utilise les concepts suivants :
    - Entité
    - Association
    - Identifiants
    - Attributs
    - Cardinalité



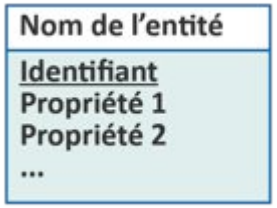
Exemple d'un MCD

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Entité et attributs :

- À partir du dictionnaire de données, on regroupe les données élémentaires par concept appelé **entité**. Une **entité** est un élément unique décrit par un ensemble de **propriétés** (aussi appelées **attributs**). Une de ces propriétés est la source des dépendances fonctionnelles avec le reste des propriétés. Elle joue le rôle d'un **identifiant** unique de l'entité.
- Le nom d'une entité est souvent un nom représentant un « objet de gestion ». Exemple : Étudiant, Formation, Article, Fournisseur…
- Une entité est formalisée comme suit :



- Une entité est aussi l'ensemble des occurrences.

Exemple d'occurrences de l'entité FORMATION :

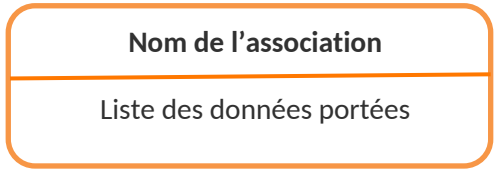
ID01	**CCP01**
Introduction au développement	**C/C++**
3	**30**
2500	**3000**

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Association :

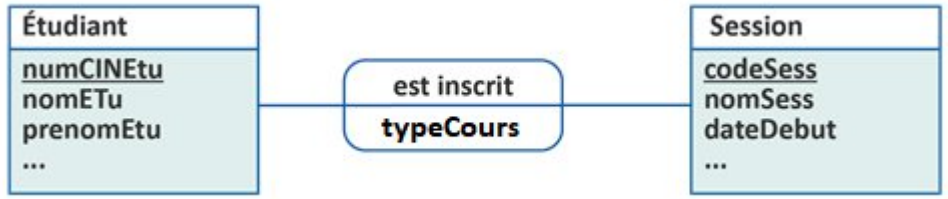
- Une association est un lien entre deux ou plusieurs entités. Ce lien est défini par des règles de gestions non traduites sous forme d'entité simple. Une association porteuse peut avoir des propriétés aussi. Une association est formalisée comme suit :



- Les données du dictionnaire de données qui dépendent de plusieurs entités sont mises dans l'association (porteuse) qui relie ces entités. Une association est dite binaire si elle relie entre deux entités, et tertiaire si elle relie entre trois entités.

Exemple :

- la règle de gestion : « Un étudiant est inscrit dans une session d'une formation » ainsi que « lors de l'inscription l'étudiant choisis le type de cours » sera représentée par l'association « est inscrit ». Cette association porte la donnée: typeCours.



Copyright - Tout droit réservé - OFPPT

Les cardinalités d'une association :

- Les cardinalités indiquent le nombre de fois où une entité est concernée par une association. Elles sont déduites des règles de gestion. Il y a trois valeurs typiques : 0, 1 et N (plusieurs).
- Les entités liées par une association possèdent chacune deux cardinalités : minimum et maximum. En effet, pour une association entre deux entités, il y aura quatre cardinalités à définir.
- Les cardinalités sont déduites à partir des règles de gestion. Ces règles sont propres à l'organisation étudiée et expriment des contraintes sur le modèle.

Remarque :

- Il se peut qu'il y ait des règles de gestions qui imposent un nombre précis de cardinalités. Ceci devra être géré par des traitements supplémentaires.

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Les cardinalités d'une association :

- Comment définir les cardinalités ?

La cardinalité minimum :

➢ = 0 si la participation des occurrences d'une entité dans l'association est facultative.

Exemple : on peut avoir des Formations qui n'ont aucune session programmée.

➢ = 1 si la participation des occurrences d'une entité dans l'association est obligatoire.

Exemple : On ne peut pas avoir une session sans qu'elle soit liée à une formation.

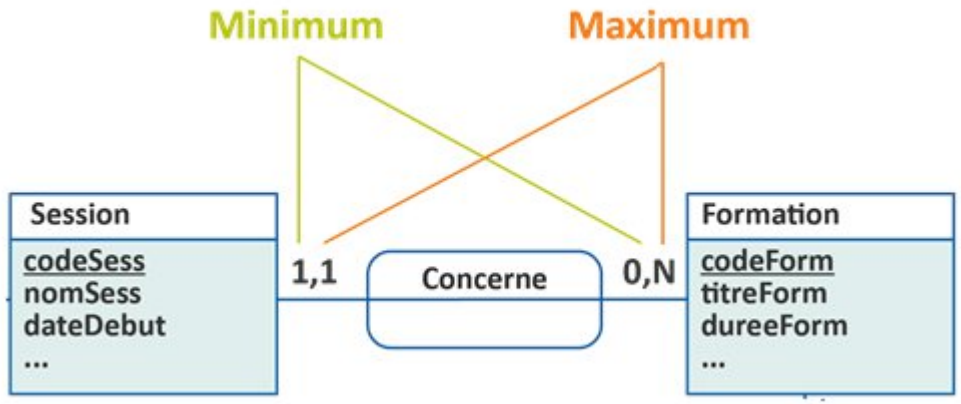
La cardinalité maximum :

➢ = 1 si la participation des occurrences d'une entité dans l'association est exclusive.

Exemple : une session concerne une et une seule formation.

➢ = N si la participation des occurrences d'une entité dans l'association est multiple.

Exemple : une formation peut avoir plusieurs sessions.



Copyright - Tout droit réservé - OFPPT

PARTIE 1

Les cardinalités d'une association :

- Voici un tableau récapitulatif des types de cardinalités les plus répandues :

Cardinalité	Signification
0, 1	**Au plus un :** chaque occurrence de l'entité n'est pas obligatoirement concernée par l'association et si elle l'est, c'est au plus une seule fois.
1, 1	**Un et un seul :** chaque occurrence de l'entité est concernée par l'association exactement une fois.
0, N	**Zéro, un ou plusieurs :** chaque occurrence de l'entité n'est pas obligatoirement concernée par l'association et si elle l'est, elle peut l'être plusieurs fois.
1, N	**Au moins un :** chaque occurrence est concerné par l'association et peut l'être plusieurs fois.

Copyright - Tout droit réservé - OFPPT

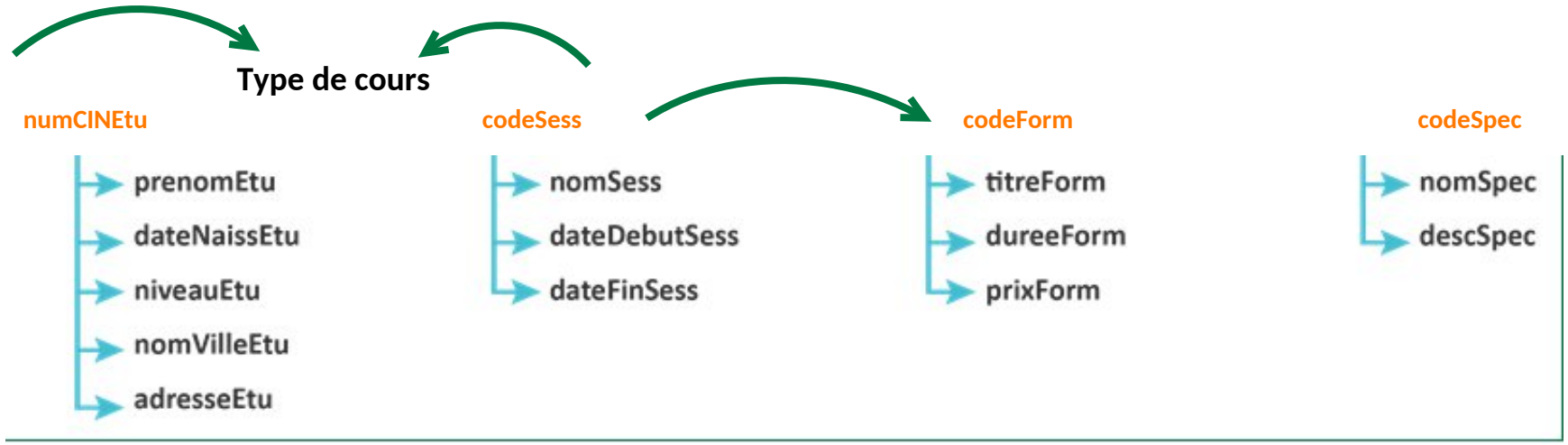
PARTIE 1

- Le passage du graphe des dépendances fonctionnelles au MCD se fait en respectant les règles suivantes :

	RÈGLES	OBSERVATIONS
N°1	Toute donnée du graphe devient une propriété.	----------
N°2	Chacune des données sources de dépendance fonctionnelle devient l'identifiant d'une entité.	----------
N°3	Une dépendance fonctionnelle entre deux données sources se traduit en association non porteuse de propriétés.	----------
N°4	Une donnée source de DF qui est relevée de l'association de plusieurs données élémentaires se traduit par une association porteuse de propriétés.	Il s'agit d'une association hiérarchique appelée aussi association fonctionnelle ou CIF (Contrainte d'Intégrité Fonctionnelle).
N°5	Des associations (issues de dépendances non fonctionnelles) peuvent exister dans un MCD sans pour autant faire partie du graphe des DF.	Il s'agit d'une association non hiérarchique appelée aussi association non fonctionnelle ou CIM (Contrainte d'Intégrité Multiple).

Copyright - Tout droit réservé - OFPPT

- Le graphe des DF permet de construire un modèle conceptuel de données (MCD) fiable. Même si les dépendances fonctionnelles sont la plupart du temps évidentes et ne nécessitent pas une représentation graphique, le graphe des DF aide toutefois à distinguer entre les futures éléments du MCD

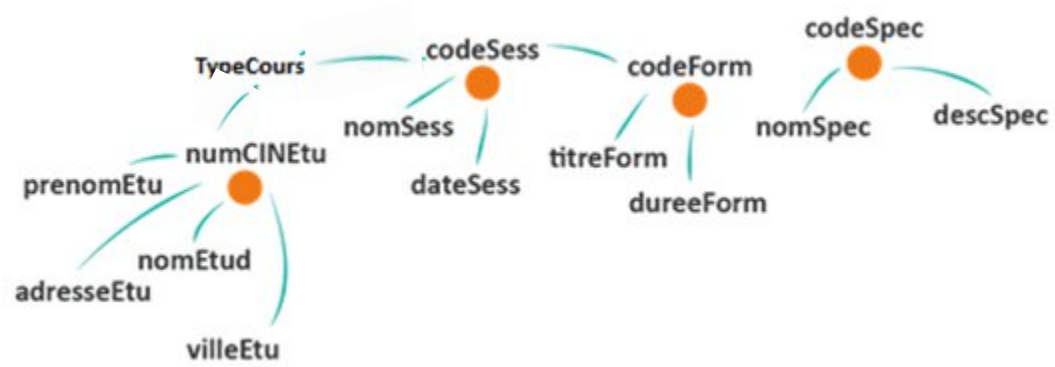


Graphe des DF de l'exemple du centre de formation

Copyright - Tout droit réservé - OFPPT

PARTIE 1

Exemple : (voir graphe des dépendances fonctionnelles ci-dessous)



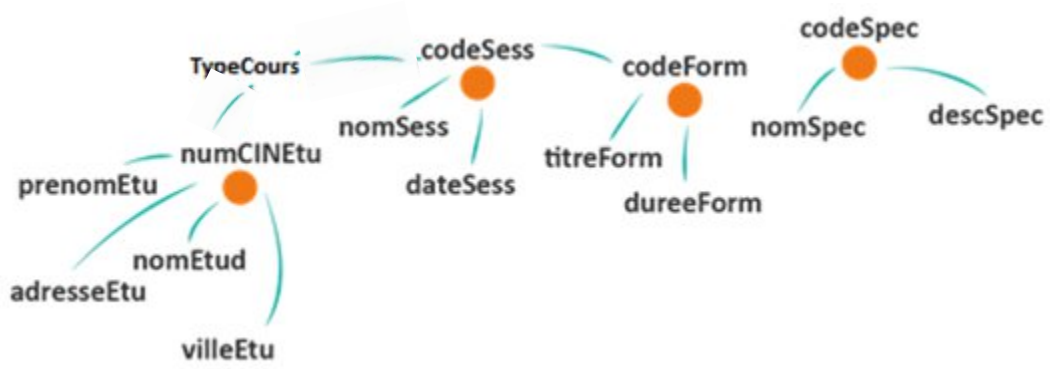
- En appliquant les règles N°1 et N°2, on peut déduire la liste des entités avec leurs propriétés et identifiants :
- **ÉTUDIANT** (NumCINETU, nomEtu, prenomEtu, adresseEtu,…)
- **FORMATION** (codeForm, titreForm, dureeForm, prixForm)
- **SESSION** (codeSess, nomSess, dateDebutSess, dateFinSess)
- **SPÉCIALITÉ** (codeSpec, nomSpec, descSpec)

	RÈGLES
N°1	Toute donnée du graphe devient une propriété.
N°2	Chacune des données sources de dépendance fonctionnelle devient l'identifiant d'une entité.
N°3	Une dépendance fonctionnelle entre deux données sources se traduit en association non porteuse de propriétés.
N°4	Une donnée source de DF qui est relevée de l'association de plusieurs données élémentaires se traduit par une association porteuse de propriétés.
N°5	Des associations (issues de dépendances non fonctionnelles) peuvent exister dans un MCD sans pour autant faire partie du graphe des DF.

Copyright - Tout droit réservé - OFPPT

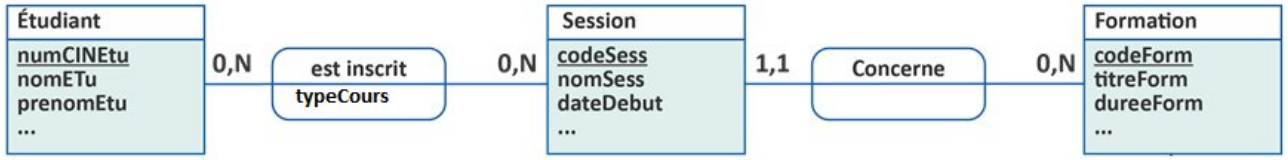
PARTIE 1

Exemple : (voir graphe des dépendances fonctionnelles ci-dessous)



- Les règles N°3 et N°4 permettent de déduire les deux associations fonctionnelles : **Concerne** (règles N°3) - **est Inscrit** (règles N°4)

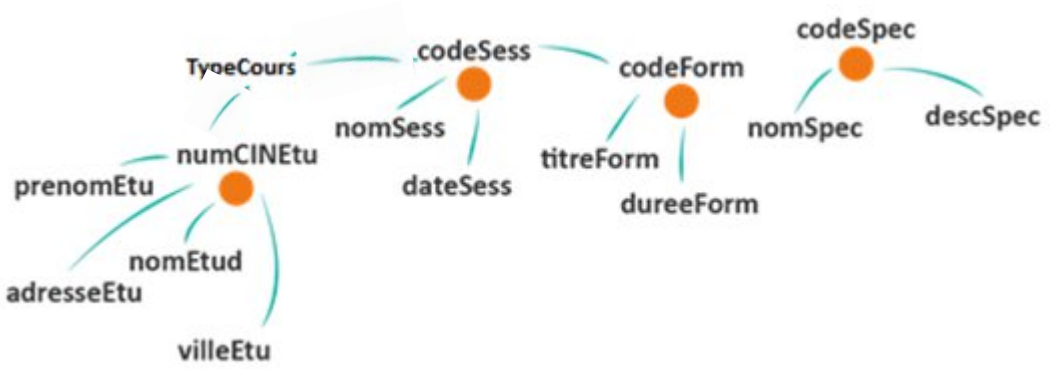
	RÈGLES
N°1	Toute donnée du graphe devient une propriété.
N°2	Chacune des données sources de dépendance fonctionnelle devient l'identifiant d'une entité.
N°3	Une dépendance fonctionnelle entre deux données sources se traduit en association non porteuse de propriétés.
N°4	Une donnée source de DF qui est relevée de l'association de plusieurs données élémentaires se traduit par une association porteuse de propriétés.
N°5	Des associations (issues de dépendances non fonctionnelles) peuvent exister dans un MCD sans pour autant faire partie du graphe des DF.



Copyright - Tout droit réservé - OFPPT

PARTIE 1

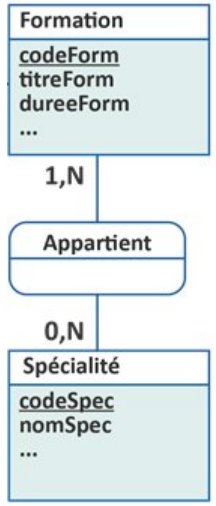
Exemple : (voir graphe des dépendances fonctionnelles ci-dessous)



En appliquant la règle N°5, nous définissons l'association non fonctionnelle entre l'entité **FORMATION** et **SPECIALITE** :

Une formation peut appartenir à plusieurs specialités,

chaque specialité peut contenir plusieurs formations)

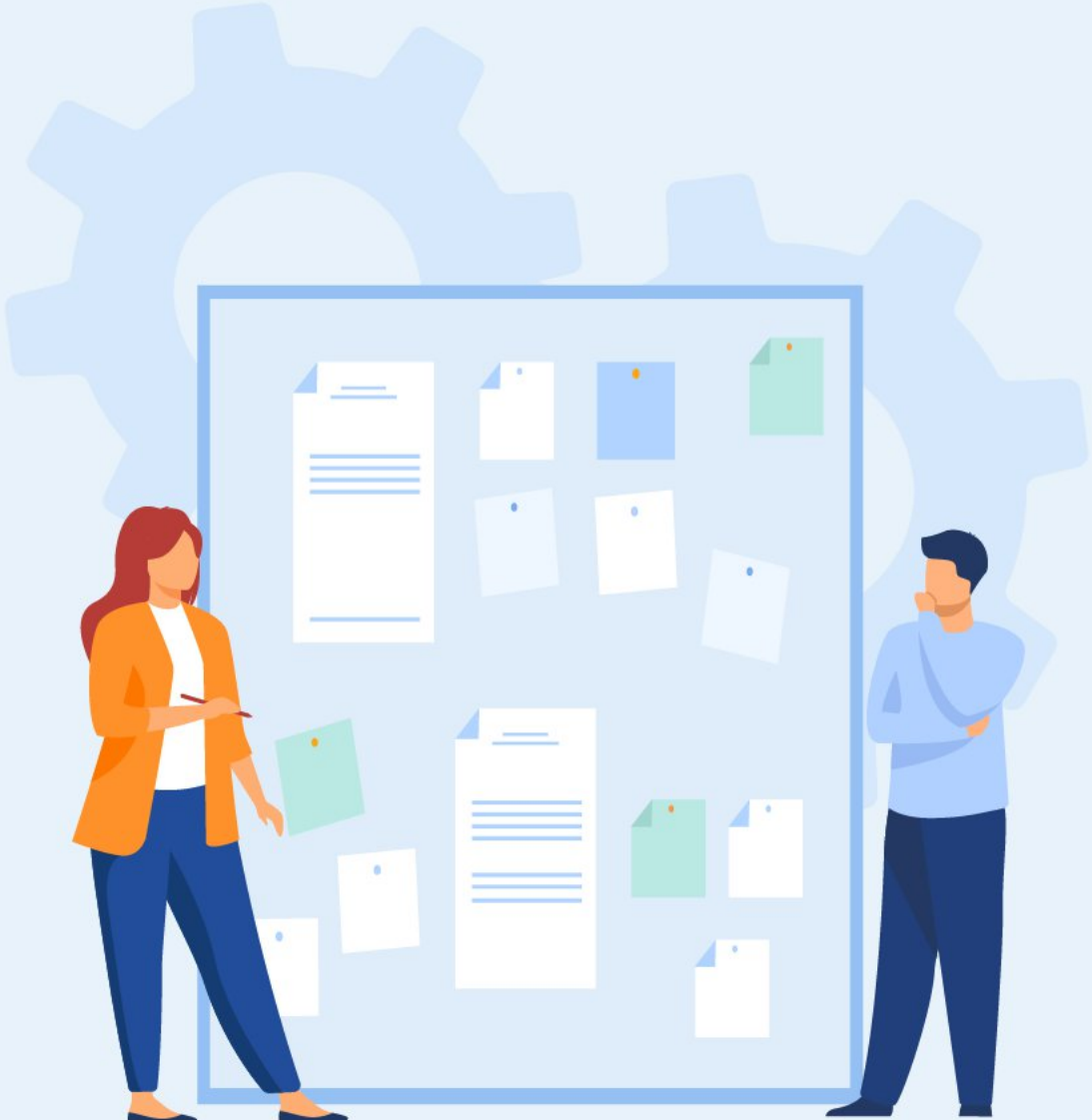


	RÈGLES
N°1	Toute donnée du graphe devient une propriété.
N°2	Chacune des données sources de dépendance fonctionnelle devient l'identifiant d'une entité.
N°3	Une dépendance fonctionnelle entre deux données sources se traduit en association non porteuse de propriétés.
N°4	Une donnée source de DF qui est relevée de l'association de plusieurs données élémentaires se traduit par une association porteuse de propriétés.
N°5	Des associations (issues de dépendances non fonctionnelles) peuvent exister dans un MCD sans pour autant faire partie du graphe des DF.

Copyright - Tout droit réservé - OFPPT

CHAPITRE 2

MODÉLISATION DES DONNÉES



1. Contraintes déduites des règles de gestion
2. Dictionnaire de données
3. Construction du graphe de dépendances fonctionnelles
4. Règles de passage du graphe au modèle conceptuel de données
5. **Construction du modèle conceptuel de données**

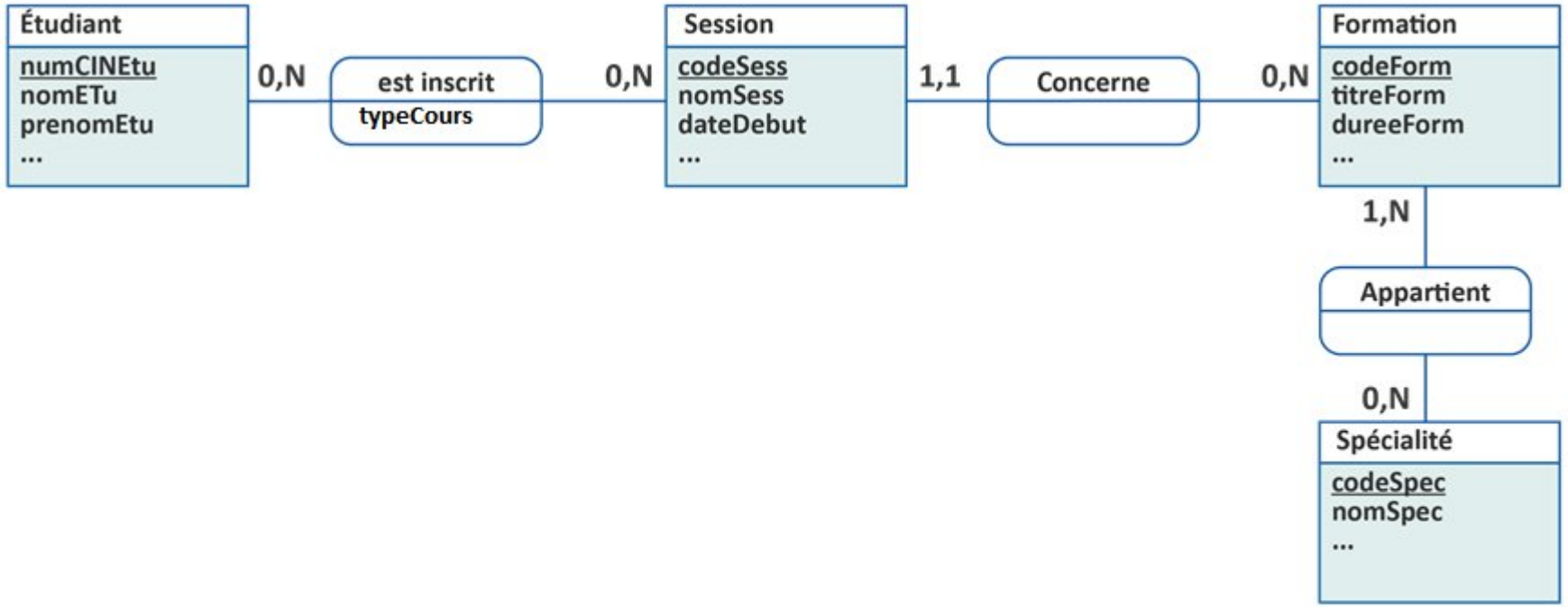
Quelle démarche pour la construction du MCD ?

- À ce stade, il est possible d'élaborer le MCD complet à partir des entités et associations ainsi que les données du dictionnaire des données, et ce en suivant la démarche suivante :
    - Afin d'assurer la validité du modèle, il faut observer les points suivants :
        - Toutes les entités du MCD doivent être reliées à, au moins, une association.
        - S'assurer de la conformité du modèle aux contraintes et règles de gestion.
        - Évaluer le modèle contre ce que les utilisateurs comprennent.
        - En cas de modèle complexe, il faut commencer par les entités et associations élémentaires puis itérer en complétant progressivement jusqu'à ce que le modèle semble raisonnablement complet.

Copyright - Tout droit réservé - OFPPT

Exemple :

- MCD correspondant au projet du «Centre de formation» :

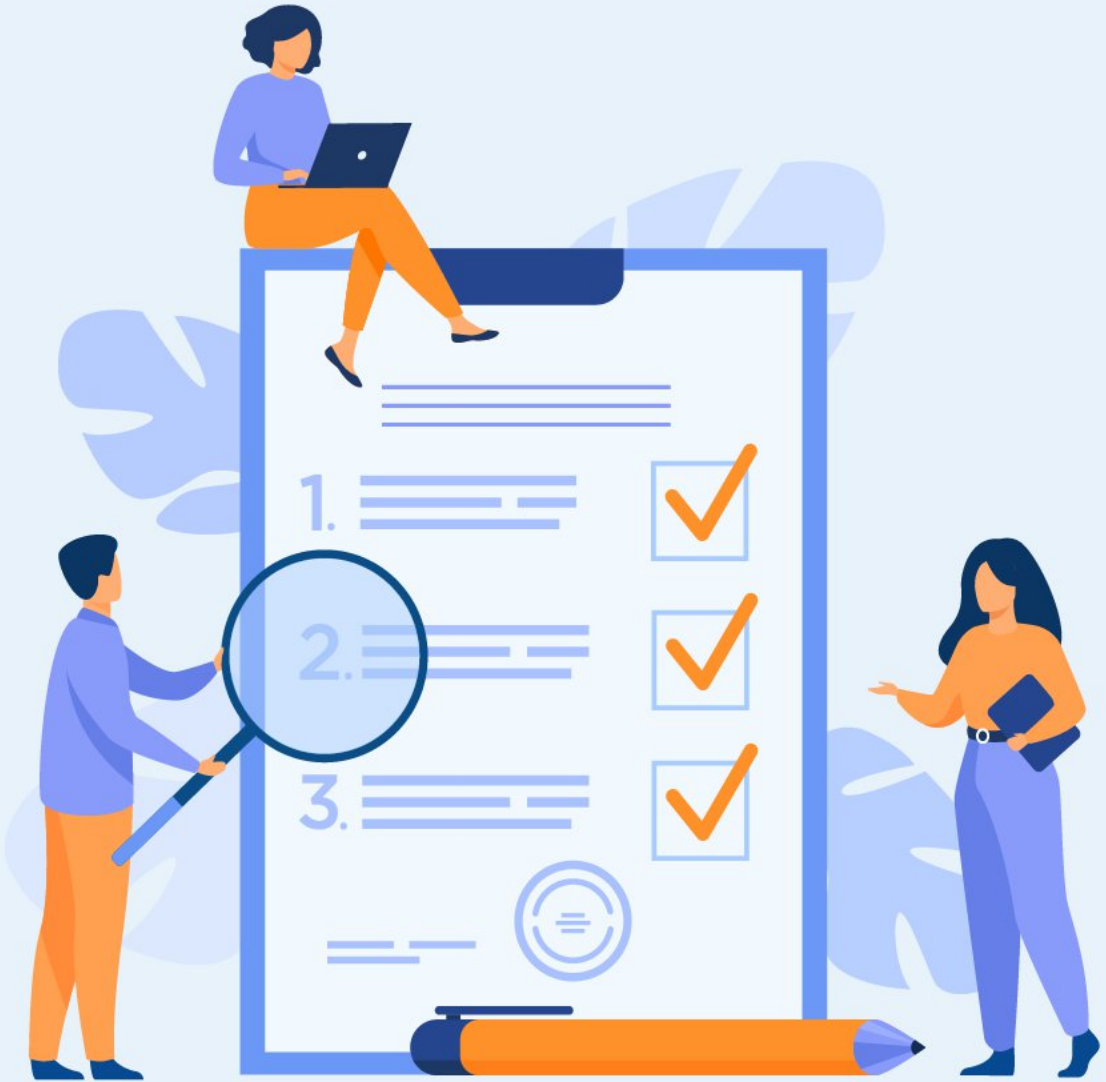


PARTIE 1

Copyright - Tout droit réservé - OFPPT

CHAPITRE 3

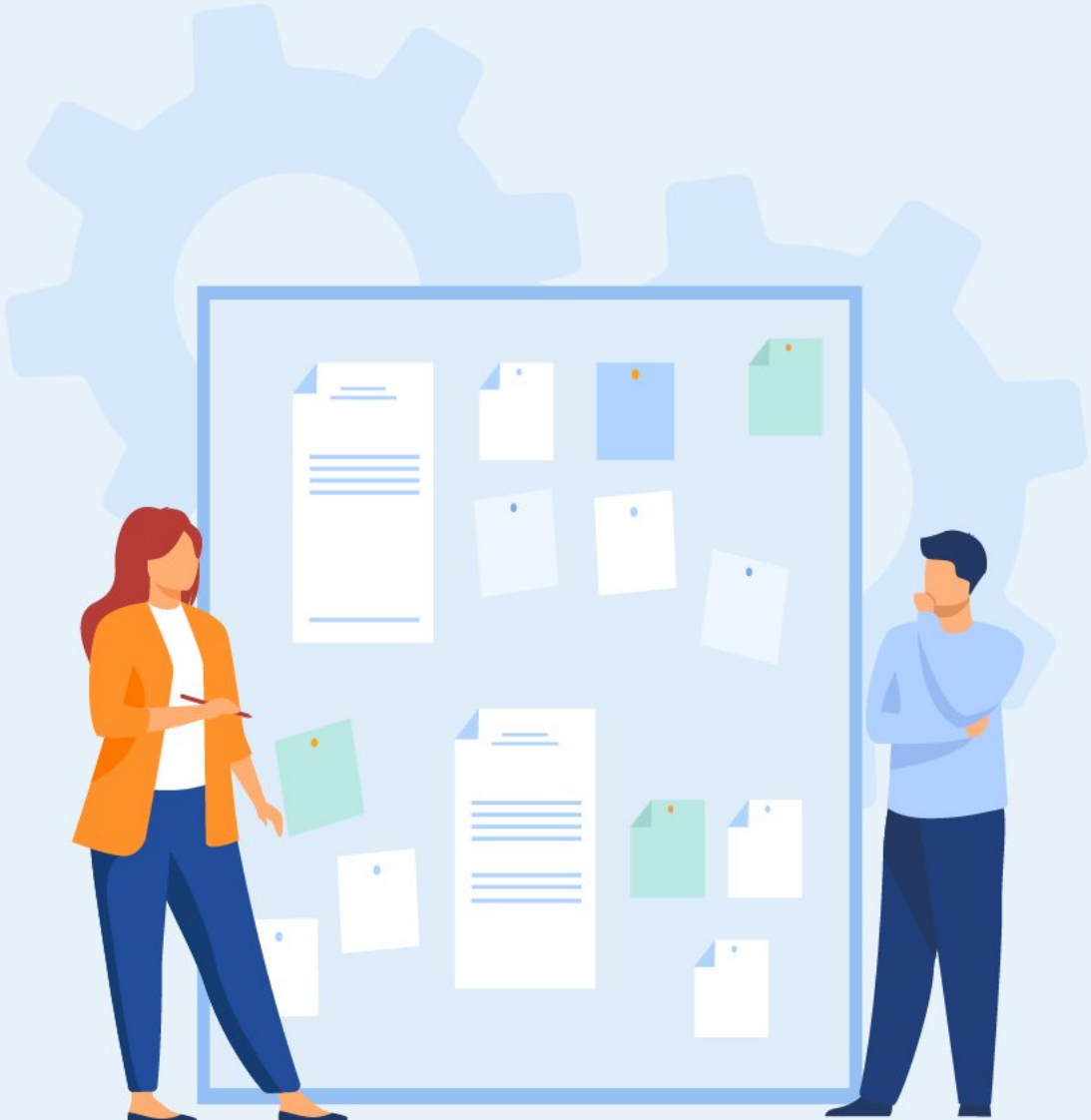
NORMALISATION DES DONNÉES



Ce que vous allez apprendre dans ce chapitre :

- L'identification des différentes formes normales
- La construction du modèle logique des données (MLD) normalisé

05 heures



CHAPITRE 3

NORMALISATION DES DONNÉES

1. **Formes normales**
2. Règles de passage du MCD au MLD normalisé

03 - Normalisation des données

Formes normales

- Les formes normales permettent la décomposition des entités en des relations, sans perdre d'informations, en se basant sur les de dépendances fonctionnelles, dans le but de construire un schéma conceptuel représentant de manière correcte les associations canoniques du monde réel.
- Au niveau de la base de données, ce travail permet d'éviter les redondances et facilite la maintenance des données.

- Il existe différents niveaux de formes normales :
  - ✓ **Première forme normale (1FN ou 1NF)**
  - ✓ **Deuxième forme normale (2FN ou 2NF)**
  - ✓ **Troisième forme normale (3FN ou 3NF)**
  - ✓ **Forme normale de Boyce-Codd (FNBC ou BCNF)**
  - ✓ **Quatrième forme normale (4FN – NF4)**

PARTIE 1

Copyright - Tout droit réservé - OFPPT

Première forme normale (1FN ou 1NF) : Attribut élémentaire

- Une relation est en première forme normale si, et seulement si, tous ses attributs sont atomiques et sont en dépendance fonctionnelle avec l'identifiant de cette relation. Ceci dit, dans un attribut, on ne peut avoir qu'une seule valeur. Un attribut est atomique s'il ne contient qu'une seule valeur pour un tuple (c'est-à-dire une ligne de données), c'est a dire qu'il ne regroupe pas un ensemble de plusieurs valeurs.

Exemple :

- **La relation :** Formation (**codeForm**, titreForm, codeSpec…) : Cette relation n'est pas en 1FN car l'attribut « codeSpec » n'est pas en dépendance fonctionnelle avec l'identifiant: codeFom

- **La relation :** Étudiant (numCINEtu, nomEtu, **prenoms**…) : Cette relation n'est pas en 1FN si on stocke plusieurs valeurs dans l'attribut « prénoms ». La forme correcte serait : Étudiant (numCINEtu, nomEtu, prenom1, prenom2, prenom3…).

numCINEtu	nomEtu	prénoms
G683909	Alaoui	Mohammed, Amine
AB123456	Hilali	Nour, laila
..		



numCINEtu	nomEtu	prénom1	prénom2
G683909	Alaoui	Mohammed	Amine
AB123456	Hilali	Nour	Laila
..			

PARTIE 1

Copyright - Tout droit réservé - OFPPT

Deuxième forme normale (2FN ou 2NF) : Dépendance fonctionnelle élémentaire

- Une relation est en deuxième forme normale si elle vérifie les deux conditions suivantes :
    - Être en 1FN.
    - Les attributs non clé dépendent de la totalité de la clé, et non d'une partie de la clé.

- Dans le cas échéant, il faut diviser la relation en plusieurs relations regroupants un groupe d'attributs qui vérifieront la dépendance entre chaque morceau de la clé et la clé entière.

Exemple :

- **La relation :** Inscription (numCINEtu, codeSess, nomEtu, villeEtu…) **n'est pas en 2FN.**
- Cette relation doit être divisée en deux :
    - Étudiant (numCINEtu, nomEtu, villeEtu…)
    - Inscription (numCINEtu, codeSess)

Copyright - Tout droit réservé - OFPPT

Troisième forme normale (3FN ou 3NF) : Dépendance fonctionnelle élémentaire directe

- Une relation est en troisième forme normale si elle vérifie les deux conditions suivantes :
  - ✓ Être en 2FN.
  - ✓ Chacun des attributs de la relation ne dépend que de la clé et non pas d'un autre attribut de la relation.
- C'est-à-dire que toutes les dépendances fonctionnelles entre la clé primaire et les autres attributs doivent être directes, et ce pour éliminer les transitivités et les dépendances entres les attributs non clé.
- Dans le cas échéant, diviser la relation en autant de relations que de dépendances entre attributs non clé.

Exemple :

- **La relation :** Formation (codeForm, titreForm, codeSpecialite, nomSpecialite) **n'est pas en 3FN.**
- Cette relation doit être décomposée en deux :
  - Formation (codeForm, titreForm, codeSpecialite#)
  - Spécialité (codeSpecialite, nomSpecialite)

Copyright - Tout droit réservé - OFPPT

Forme normale de Boyce-Codd (FNBC ou BCNF)

- Une relation est en FNBC si elle vérifie les deux conditions suivantes :
  - ✓ Être en 3FN.
  - ✓ Les seules dépendances fonctionnelles élémentaires existantes dans les relations sont celles de la clé vers les attributs non clés.

- Cette règle permet d'éliminer les redondances créées par des dépendances entre parties de clés ainsi que celles déjà éliminées par la 3FN.

Exemple :

- **La relation :** Commune (commune, ville, région, population) **n'est pas en FNBC.**
  - Si "commune + ville" déterminent la région et la population, on a aussi ville qui détermine région.
  - Donc, on doit décomposer cette relation en :
    - Commune (commune, ville, population)
    - Ville (ville, région)

Copyright - Tout droit réservé - OFPPT

Quatrième forme normale (4FN – NF4)

- Les trois premières formes normales se focalisent sur des aspects très conceptuels et évidents, mais ne permettent pas d'éliminer toutes les redondances.
- On fait alors recours aux formes 4FN et 5FN pour ajouter une dimension de traitement de l'information et faciliter la mise à jour des données de la base.

- Une relation est en quatrième forme normale lorsque elle vérifie les deux conditions suivantes :
    - Être en 3FN.
    - Si, et seulement si, les dépendances **multi-valuées** élémentaires sont celles dans lesquelles une clé détermine la valeur d'une colonne.

Copyright - Tout droit réservé - OFPPT

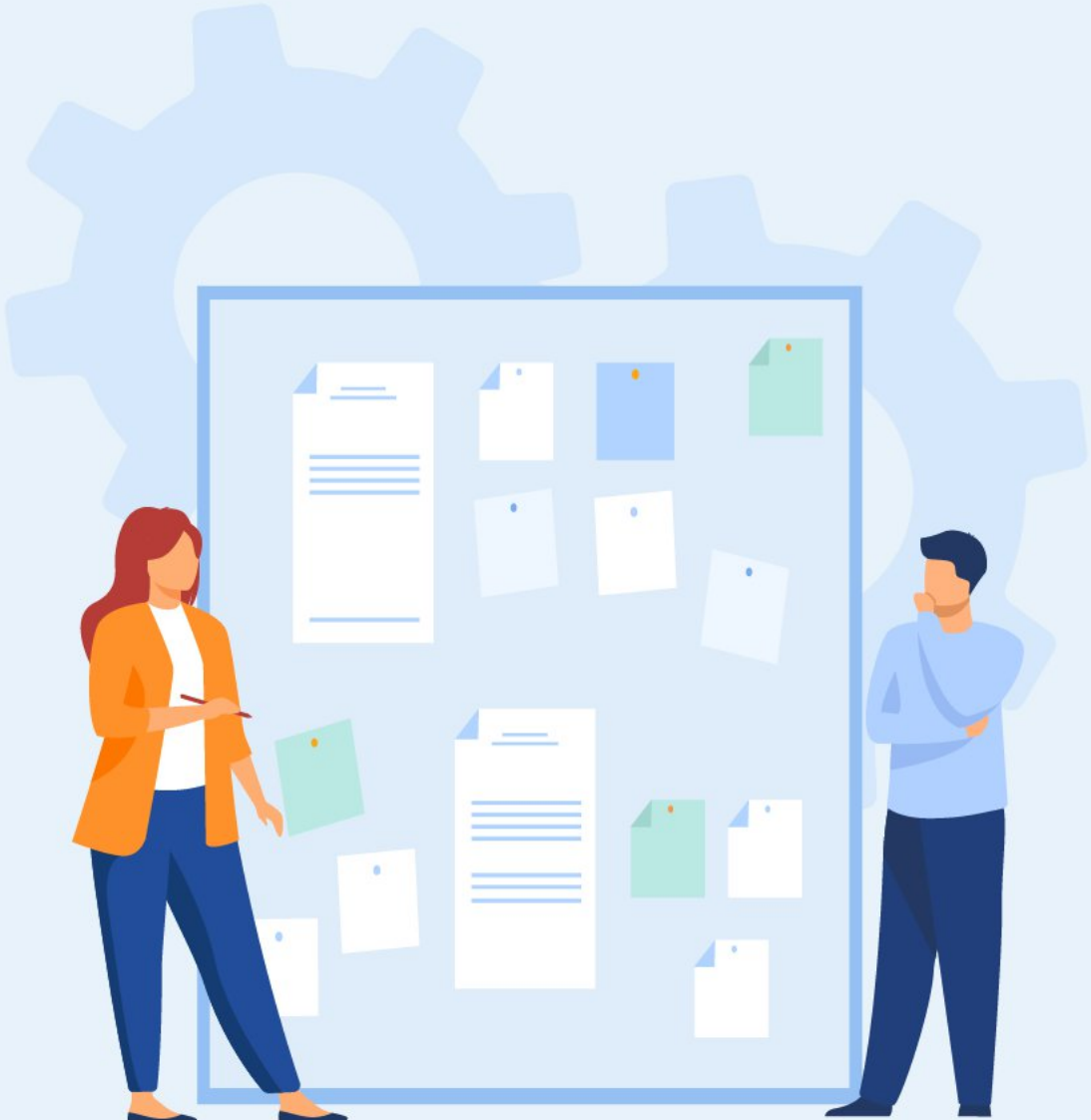
Exemple : 4FN

- **Dépendances multivaluées :** Voici la liste des modèles, couleurs et versions disponibles d'une voiture :

Modèle	Couleur	Version
715	Gris	Enjoy
715	Gris	Excellence
715	blanc	Enjoy
620	noir	Enjoy
620	blanc	Enjoy
620	blanc	Excellence

- Pour le même modèle d'une voiture, il peut exister plusieurs couleurs et plusieurs versions. La table ci-dessus illustre toute les combinaisons possibles pour chaque modèle en terme de couleur et version. En effet, la dépendance entre Modèle et Couleur d'une part, et Modèle et Version d'autre part est dite dépendance multivaluée.
- La table ainsi modélisée "Disponibilité(modèle, couleur, version)" présente un inconvénient majeur : si pour un modèle X, on veut supprimer une valeur de la colonne version V, il faudra parcourir et supprimer toutes les combinaisons ou modèles = X et Version = V. Le recours à la forme 4FN permet d'éviter ce genre de problèmes et ainsi on doit décomposer la table en deux relations :
    - DispoCouleur : (Modèle, Couleur)
    - DispoVersion : (Modèle, Version)

Copyright - Tout droit réservé - OFPPT



CHAPITRE 3

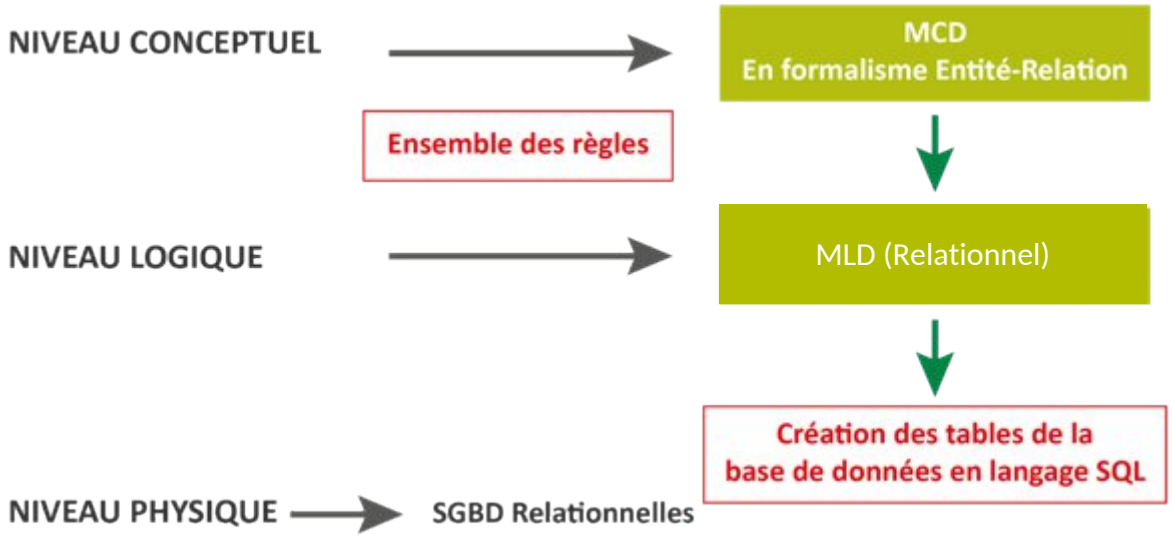
NORMALISATION DES DONNÉES

1. Formes normales
2. **Règles de passage du MCD au MLD normalisé**

Modèle Logique de Données (MLD) :

- Le modèle logique de données est une représentation du modèle de données en tables logiques reliées entre elles par des flèches. Il permet de modéliser la structure de la base de données à partir du MCD et est adapté au Systèmes de Gestion de Bases de Données Relationnelles (SGBDR).

- **Du MCD AU MLD :**



Copyright - Tout droit réservé - OFPPT

PARTIE 1

Règles de passage du MCD au MLD normalisé

- **Règle N°1 :** transformation des entités.
- **Règle N°2 :** transformation d'une association sans propriété du type (* ,n)-(1,1).
- **Règle N°3 :** transformation d'une association (1, n) -(*, n).
- **Règle N°4 :** associations ternaires (n-aires).

Règle N°1 : transformation des entités

- Une entité du MCD devient une table portant le même nom.
- Chaque ligne de la table correspond à un enregistrement.
- Chaque colonne correspond à un attribut.
- L'identifiant devient la clé primaire de la table.

Exemple :

- **ÉTUDIANT** (numCINEtu ,prenomEtu, dateNaissEtu, niveauEtu,nomVilleEtu, AdresseEtu)



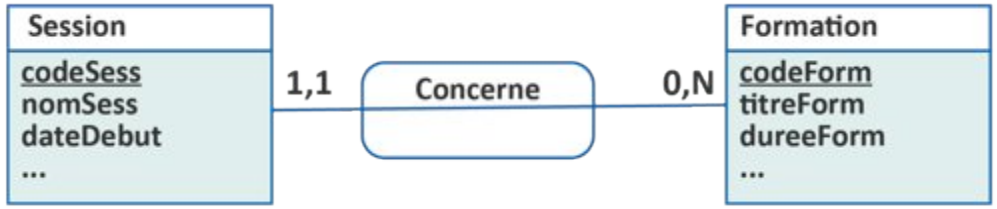
Étudiant
numCINEtu nomEtu prenomEtu dateNaissEtu niveauEtu nomVilleEtu AdresseEtu

Copyright - Tout droit réservé - OFPPT

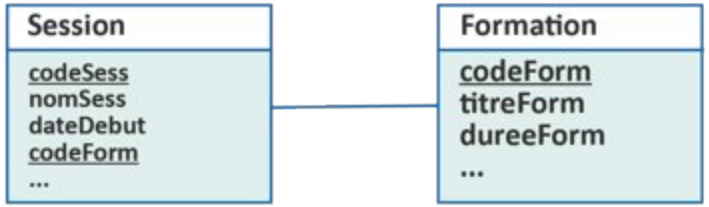
Règle N°2 : transformation d'une association sans propriété du type (* ,n)-(1,1)

- La clé primaire de la table, ayant la cardinalité (*,N), est dupliquée dans la table ayant la cardinalité (1,1).

Exemple :



- Une session concerne une seule formation.
- Une formation peut n'avoir aucune session.
- Formation est dite entité **forte** et Session est dite entité **faible**.
    - **La clé primaire** codeForm de la table Formation doit être dupliquée dans la table Session.
    - **Formation** (codeForm)
    - **Session** (codeSess)



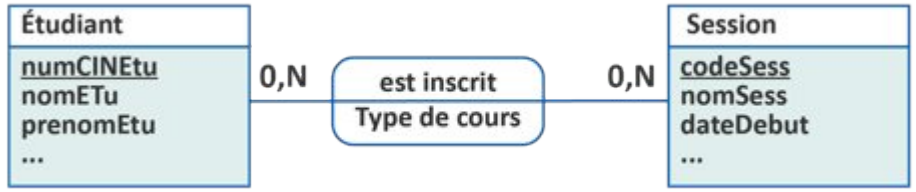
PARTIE 1

Copyright - Tout droit réservé - OFPPT

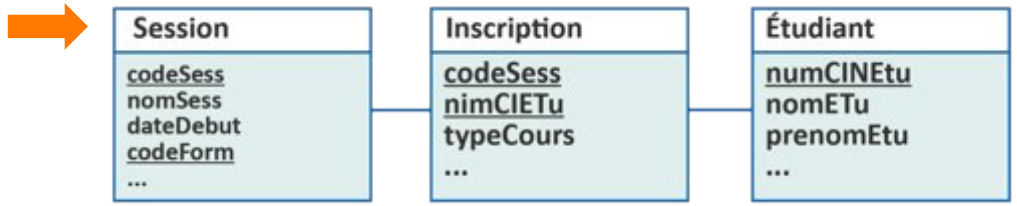
Règle N°3 : transformation d'une association (1, n) - (*, n)

- Concerne les relations ou les cardinalités max des deux côtés de l'association = N.
- La relation est transformée en une entité.
- La clé primaire de cette entité est la combinaison des clés des relations correspondantes aux entités de part et d'autre de la relation.
- Les propriétés de l'association deviennent des attributs de l'entité.

Exemple :



- Un étudiant peut n'être inscrit à aucune session de formations comme il peut être inscrit à plusieurs.
- Une session peut n'avoir aucun étudiant inscrit, comme elle peut avoir plusieurs étudiants inscrits.
- L'association « est inscrit » est transformée en une nouvelle entité « Inscription ».
  - **Inscription** (codeSess, numCINEtu, typeCours)



PARTIE 1

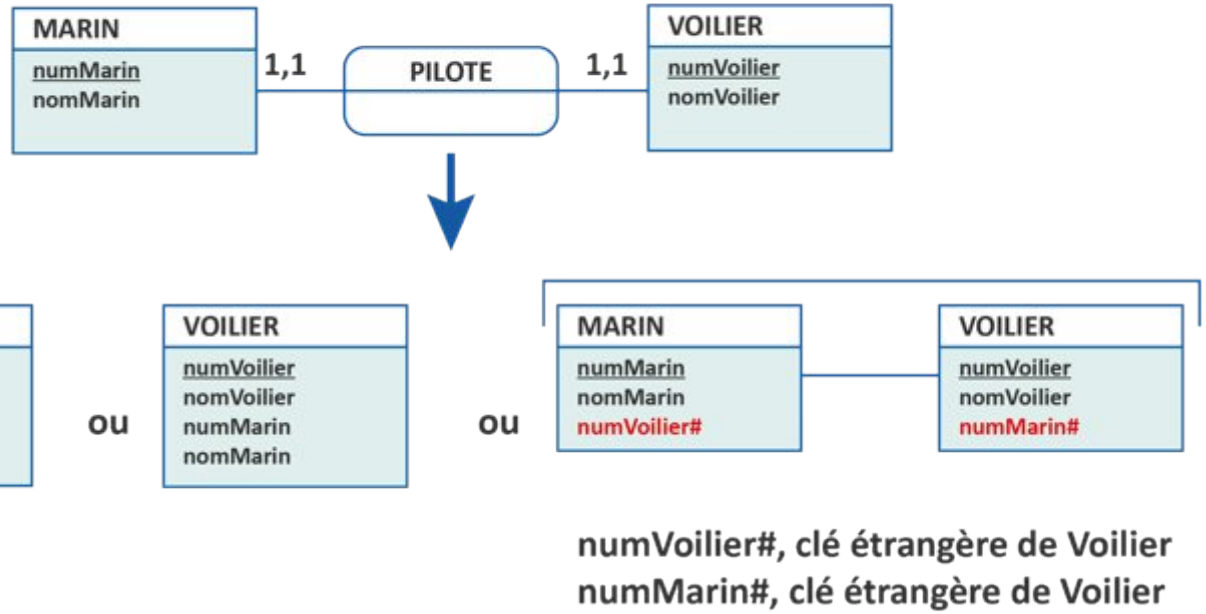
Copyright - Tout droit réservé - OFPPT

Règle N°3 : transformation d'une association (1, n) - (*, n)

Cas particuliers :

- Associations 1,1

Exemple : course à la voile : 3 solutions



numVoilier#, clé étrangère de Voilier
numMarin#, clé étrangère de Voilier

PARTIE 1

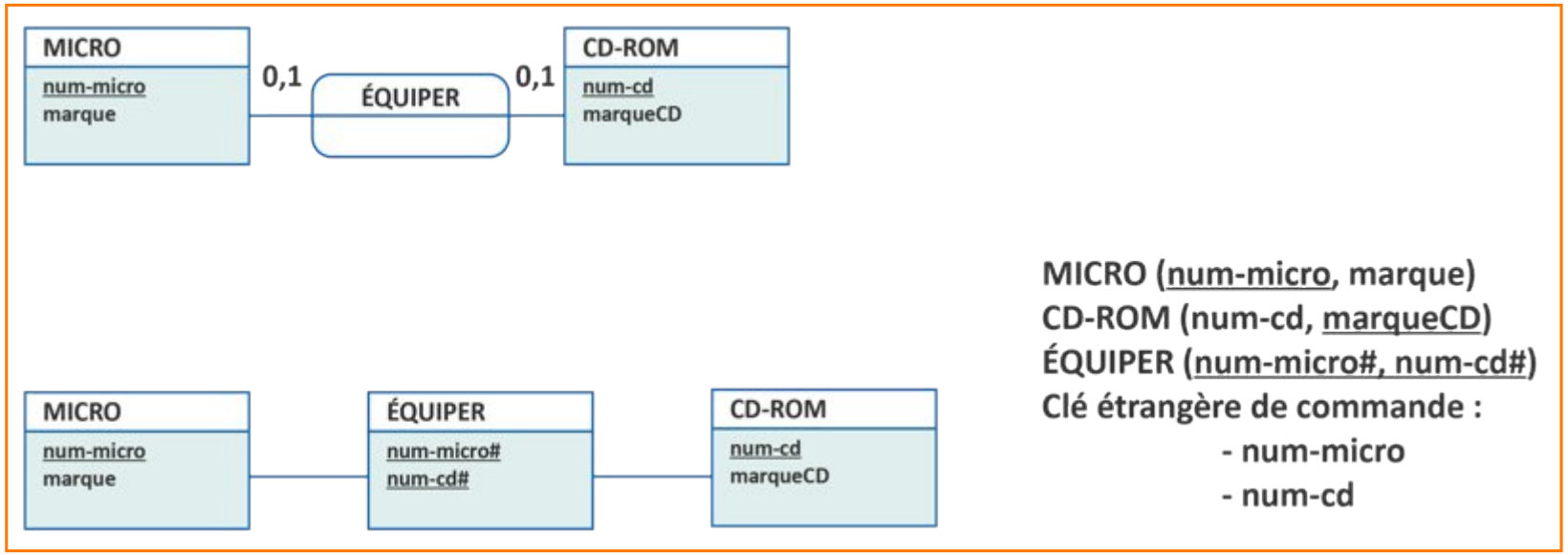
Copyright - Tout droit réservé - OFPPT

PARTIE 1

Règle N°3 : transformation d'une association (1, n) - (*, n)

Cas particuliers :

- Associations binaires 01,01



Copyright - Tout droit réservé - OFPPT

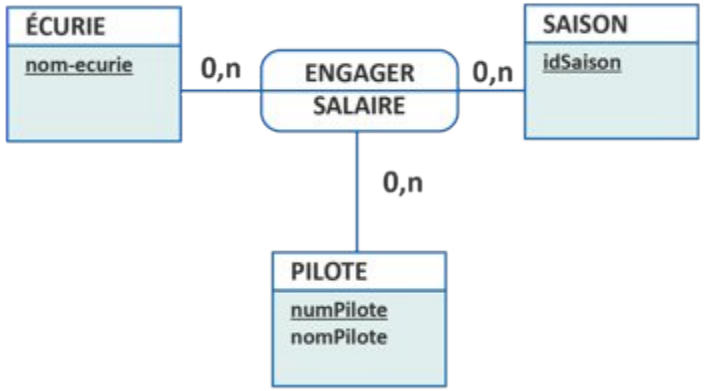
PARTIE 1

Règle N°4 : associations ternaires (n-aires)

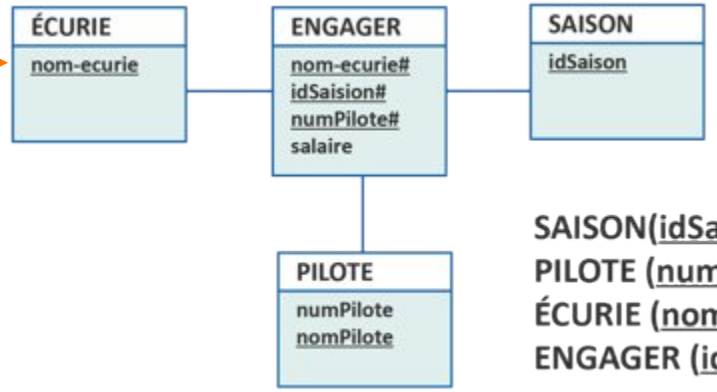
- L'association gère une table, qui reçoit en clé étrangère, les clés primaires des tables associées.
- La composition des clés étrangères devient la clé primaire de la table association.
- Les données éventuelles de l'association deviennent les attributs de la table association.

Exemple :

- **L'association suivante :**



Est transformée comme suit :

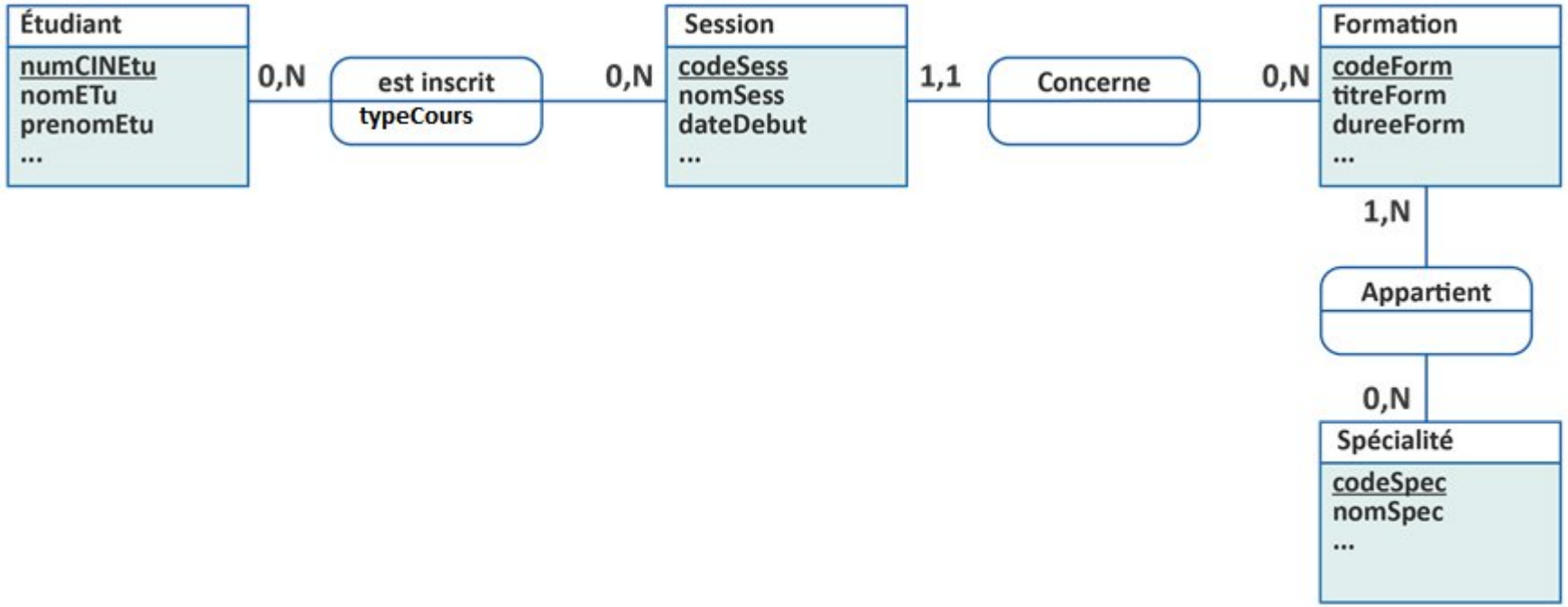


SAISON(idSaison)
PILOTE (numPilote, nomPilote)
ÉCURIE (nom-ecurie)
ENGAGER (idSaison, numPilote, nom-ecurie, salaire)

Copyright - Tout droit réservé - OFPPT

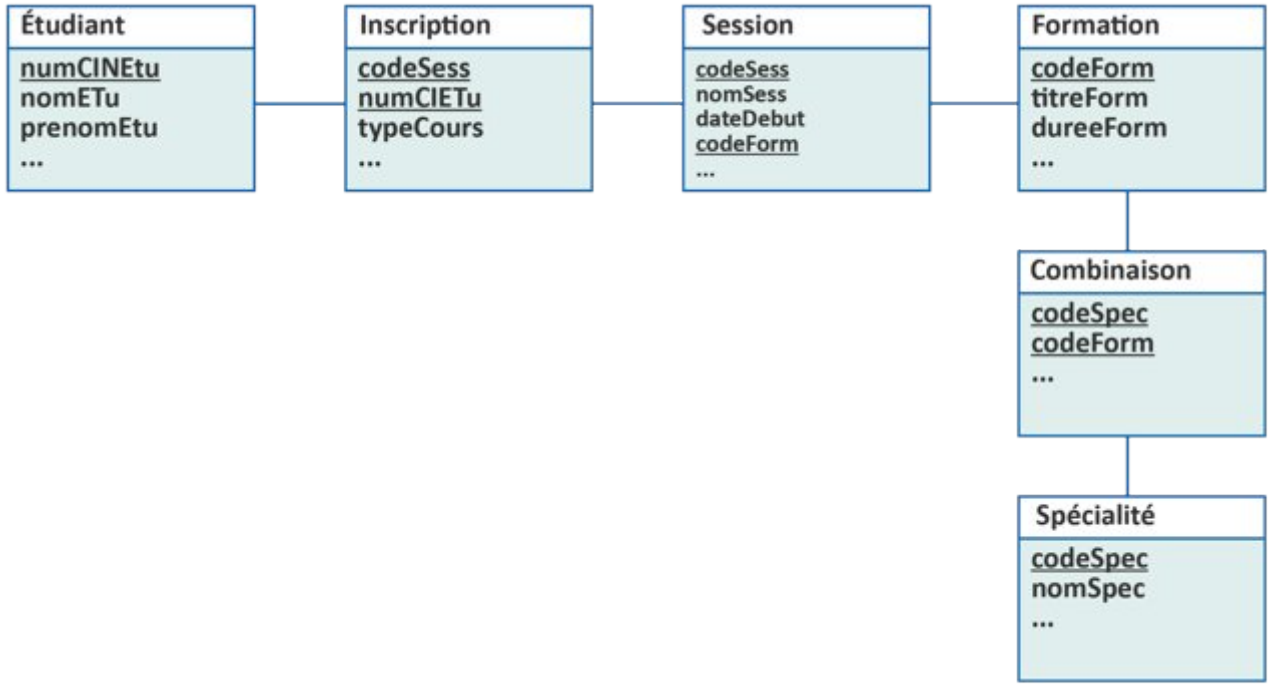
PARTIE 1

Le MCD de la base de données du centre de formation



Copyright - Tout droit réservé - OFPPT

Le MLD correspondant au MCD du centre de formation



Remarques :

- **Deux nouvelles entités ajoutées :** Inscription et Combinaison identifiées respectivement pas les clés (codeSess,numCINEtu) et (codeSpec,codeForm).
- **L'association entre Session et Formation est traduite par l'ajout de la clé :** codeForm dans l'entité Session.

PARTIE 1

Copyright - Tout droit réservé - OFPPT