



Bases de données : Merise (modèle des données)

Support de Formation n°2



Mise à jour 3 mai 2001 (LR, MF)

Objectifs

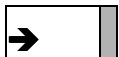
- Situer les modèles conceptuel et logique des données dans la méthode Merise.
- Créer des modèles conceptuel de données à partir de cas concrets.
- Convertir le MCD en modèle logique de données.
- Créer la base à partir du modèle logique.

Mode d'emploi

Pour chacun des thèmes abordés, ce support de formation :

- présente les points essentiels
- renvoie à des supports de cours par le symbole : 
- propose des exercices par le symbole : 

Autre symbole utilisé :



Approfondissement : travail de documentation ou exercice supplémentaire, à faire individuellement (non intégré dans le temps de formation moyen)

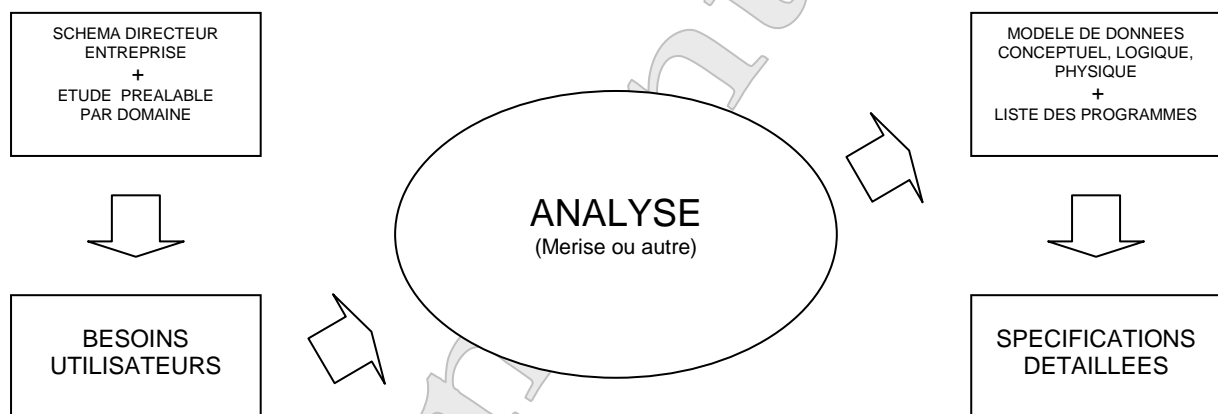
1. PRESENTATION DE LA METHODE MERISE

1.1 Ressources

- Le document AFPA « Merise 1 : Modèle de données Conceptuel, Logique, Physique » du CFPA de Toulouse est référencé dans ce guide de formation.
- Les documents « Merise 2 à 6 » qui portent sur les modèles des traitements seront distribués à titre de documents d'autoformation.

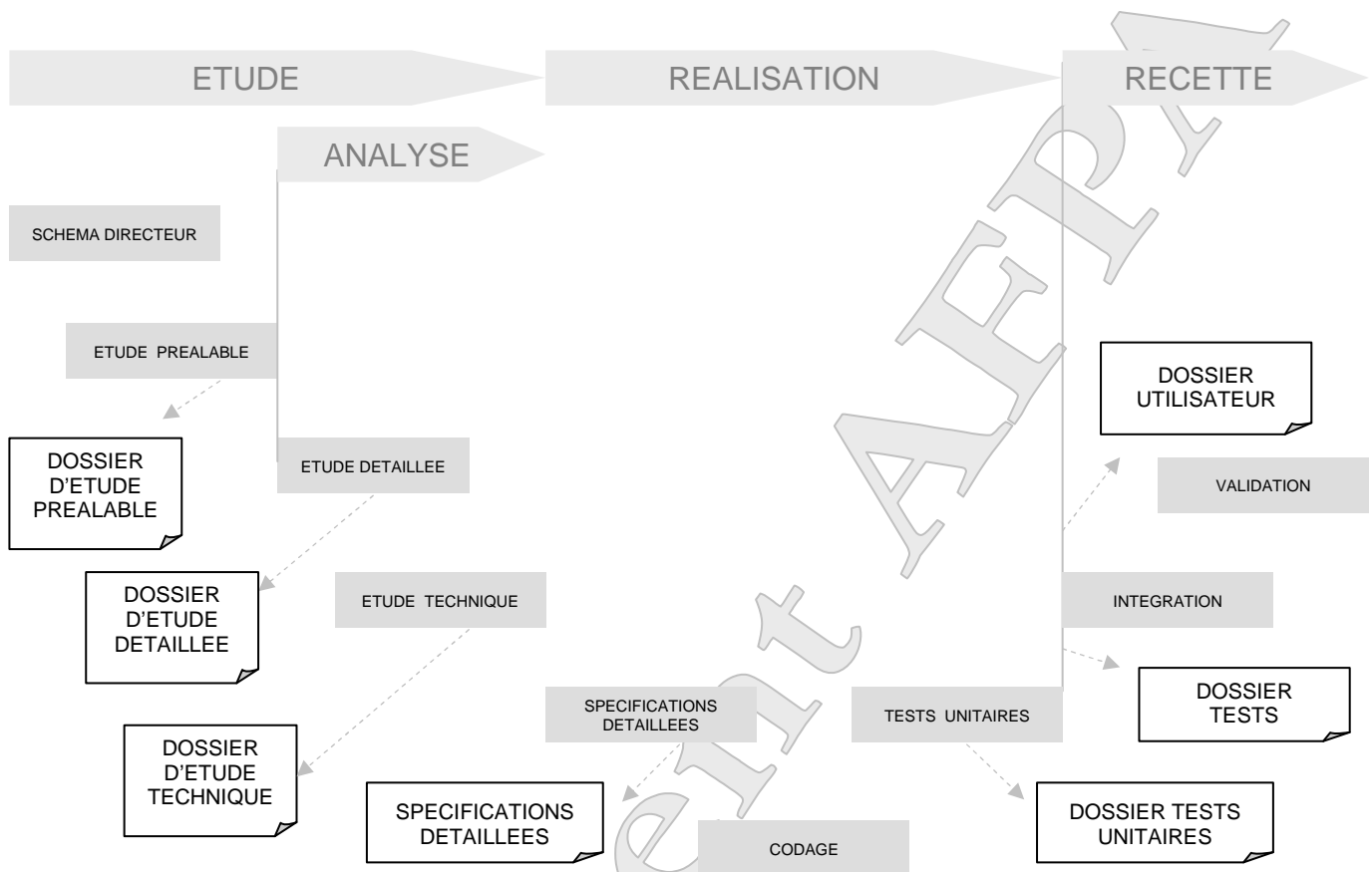
1.2 Comment se situer dans la méthode Merise

- En matière d'analyse, la méthode MERISE est la plus utilisée en France. Elle prend en compte le besoin exprimé par le client (son domaine d'information) pour en élaborer l'informatisation, ce qui peut se résumer par ce schéma :



- Les phases amont ne concernent qu'indirectement les développeurs. Le schéma directeur décrit l'organisation générale de l'entreprise et les grands flux de données : il est défini par la direction informatique de l'entreprise, parfois avec l'aide de consultants externes.
- Le schéma directeur permet de situer les grands « **domaines** » d'application de l'entreprise : comptabilité, production, gestion du personnel, marketing... Chaque domaine fait ensuite l'objet d'une « **étude préalable** » qui permet d'isoler toutes les fonctionnalités d'un domaine et de définir les applications nécessaires.
- Remarquons que Merise est d'abord une méthode d'organisation : toutes les fonctionnalités d'un domaine n'aboutissent pas nécessairement à un projet informatique.
- Pour chaque fonctionnalité à informatiser, on réalise un dossier « **d'étude détaillée** » qui référence précisément les besoins des utilisateurs pour une application déterminée. Cette tâche est souvent confiée à un chef de projet, mais elle peut aussi être attribuée à un développeur dans une PME, ou pour un petit projet.
- Dans le premier projet de la formation « Développeur Informatique », vous partirez d'un dossier « **d'étude détaillée** » rédigée par nous, et vous aurez à réaliser le dossier « **d'étude technique** »,

et toutes les phases suivantes. Dans le projet final, vous devrez réaliser vous-mêmes les modèles de données de l'étude détaillée.



- Dans la pratique, on commence l'étude détaillée en recensant les besoins des utilisateurs, par une série d'entretiens avec les clients.
- Un des principes fondamentaux de la méthode Merise consiste à bien séparer l'analyse des données de l'analyse des traitements : à l'issue de chaque entretien, il faudra donc faire des fiches qui résument les traitements demandés et qui décrivent les données de l'entreprise.
- Cette démarche préalable n'est pas évidente : les clients ne connaissent pas les principes de Merise et donnent souvent « tout en vrac », données et traitements. Chaque client met l'accent sur les données ou sur les traitements, en fonction de sa culture d'origine : prépondérance du point de vue des traitements en informatique industrielle et scientifique ; prépondérance des données en informatique de gestion.
- En ce qui concerne les données, la première tâche consistera toujours à établir un « **dictionnaire de données** », où chaque donnée de l'entreprise est définie clairement par son **type** et sa **signification**, et référencée de façon unique : il faut donc détecter dans le discours du client les homonymes qui désignent en fait des données distinctes, et les synonymes qui se réfèrent à la même donnée sous des noms différents...

1.3 Mise en pratique de Merise dans ce document

- Pour se situer au plus près de la démarche d'entreprise, nous travaillerons dans ce document avec des cahiers des charges succincts, où il restera parfois des ambiguïtés : vous aurez à poser toutes les questions nécessaires au formateur, qui jouera le rôle du client.
- Merise est une méthode de communication : avant de soumettre vos modèles au formateur, vous les ferez d'abord relire par un autre groupe. Il peut aussi être utile de comparer plusieurs modèles et de chercher à comprendre les différences.

1.4 Présentation succincte des différents modèles Merise

- Merise procède :
 - du général au particulier, de l'abstrait au concret
 - par étape : étude préalable, étude détaillée, étude technique

Etude Préalable	M.C.D + M.C.T (ébauche)
Etude Détaillée	M.C.D + M.C.T (exhaustif) M.L.D + M.O.T
Etude Technique	M.P.D + MPe.T

- Certaines phases constituent une **ébauche** de la phase suivante :
 - l'étude préalable sur un domaine comprend un **MCD (Modèle Conceptuel de Données)** et un **MCT (Modèle Conceptuel des Traitements)** généraux, qui permettent de déterminer les applications qui devront être réalisées dans ce domaine d'activité ;
 - chaque application fait ensuite l'objet d'une étude détaillée qui repart des modèles précédents, pour construire un modèle des données et un modèle des traitements exhaustifs.
- Merise distingue les modèles de données et de traitements en « **Conceptuel** », « **Logique** » ou « **Organisationnel** », « **Physique** » ou « **Technique** », selon le point de vue adopté par l'analyse et le niveau de préoccupation :
 - le niveau conceptuel référence les données de l'entreprise et détermine ce qu'il faut faire, sans se préoccuper des acteurs qui utilisent les données et qui lancent les traitements, sans dire quand et où ils sont lancés... ;
 - le niveau organisationnel ou logique complète le modèle conceptuel en précisant les circonstances (Qui, Où, Quand) des actions et de l'utilisation des données ;
 - le niveau technique ou physique complète le modèle logique, en précisant les modalités de réalisation des fonctionnalités et les détails techniques portant sur le stockage des données (la fonctionnalité doit-elle être informatisée ou non ? Avec quels moyens matériels et logiciels ? S'agit-il d'une fonctionnalité de type BATCH qui sera lancée automatiquement par le système ou d'une application interactive ? Les données sont-elles gérées par un système de gestion de fichiers classique ou par un SGBDR ?)

- Les questions posées et les points de vue adoptés par les différents modèles :

Conceptuel	Quoi ?
Organisationnel (Logique)	Qui ? Où ? Quand ?
Technique (Physique)	Comment ? Quels outils ?

- Les modèles s'enchaînent rigoureusement :

NIVEAUX D'ANALYSE	DONNEES	TRAITEMENTS
Conceptuel	① M.C.D	① M.C.T
Logique	② brut M.L.D	③ M.O.T
Physique	④ optimisé ⑤ M.P.D Script SQL de création de base	⑥ MPe.T

- M.O.T** : modèle organisationnel des traitements
- M.P.D** : modèle physique des données
- MPe.T** : modèle physique externe des traitements

2. LE MODELE CONCEPTUEL DE DONNEES (MCD)



Merise I : Chap. II, construction du MCD d'une entreprise de distribution

Base de connaissance

- Le système d'information est constitué de données très diversifiées, intéressant tous les domaines, qu'il convient dans un premier temps de collecter, rassembler, trier, puis synthétiser.
- Cette étude de l'existant repose sur :



Les documents entrant et sortant des postes de travail



L'interview des utilisateurs du système, repris et rédigé précisément



Le cahier des charges

- La méthode MERISE permet de construire une représentation des données, indépendante des traitements à mettre en œuvre, en 6 étapes :

Etape 1 : le dictionnaire des données.

Etape 2 : les entités

Etape 3 : les relations simples.

Etape 4 : les relations porteuses de propriétés.

Etape 5 : les cardinalités

Etape 6 : le M.C.D final

Etape 1 : le dictionnaire des données

Données collectées

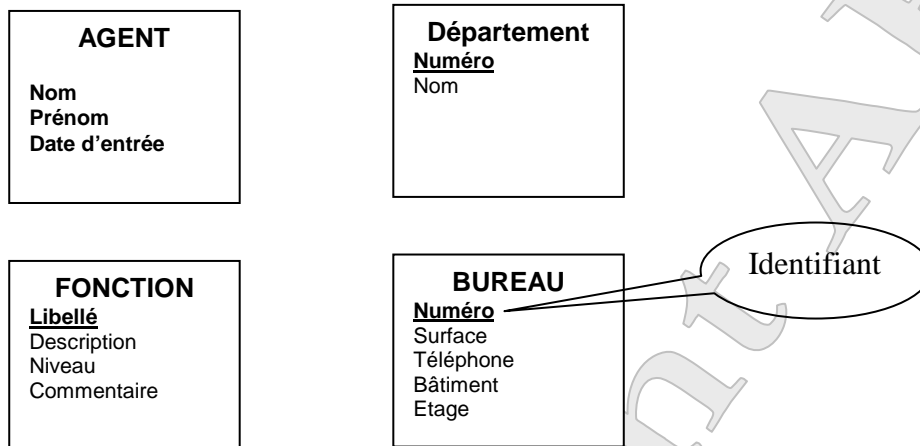
- N° de bâtiment
- Etage
- N° de département
- Nom département
- N° de bureau
- Téléphone bureau
- Surface bureau
- Nom
- Prénom
- Date d'entrée
- Libellé fonction
- Description fonction
- Niveau fonction (cadre / non cadre)
- Commentaire

On ajoutera à cette liste des données qui, bien que n'apparaissant pas clairement dans les documents consultés, concernent des paramètres qu'il faudra également gérer :

- *Type d'agent (Chef / Employé)*
- *Type fonction (Principale / Annexe)*

Etape 2 : recherche des entités

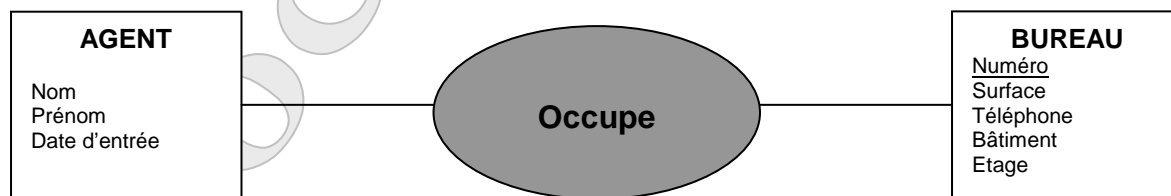
- On rassemble au sein d'une entité, toute propriété qui s'y rapporte **de façon invariante**. Ne sont donc pas propriété d'entité :
 - celles se rapportant à la fois à plusieurs entités, donc dépendant d'une relation ;
 - celles qui sont propres à une seule entité, mais qui contiennent une donnée évoluant dans le temps, si on gère l'historique.
- Les entités seront désignées par des noms communs (Agent, Département...)



- Notion « d'identifiant » ou « clé primaire » : c'est une colonne ou un ensemble de colonnes qui référencent de façon unique chaque nuplet (par exemple, un Département est parfaitement identifié par son Numéro, une Fonction par son Libellé...). Lorsque les identifiants apparaissent dans le cahier des charges, on peut les souligner dès le MCD : tous les identifiants devront de toute façon être définis au moment de passer du modèle conceptuel au modèle logique des données.

Etape 3 : Les relations simples entre entités

- Elles sont désignées par un verbe d'action que l'on choisit pour définir au mieux le sens de la relation :

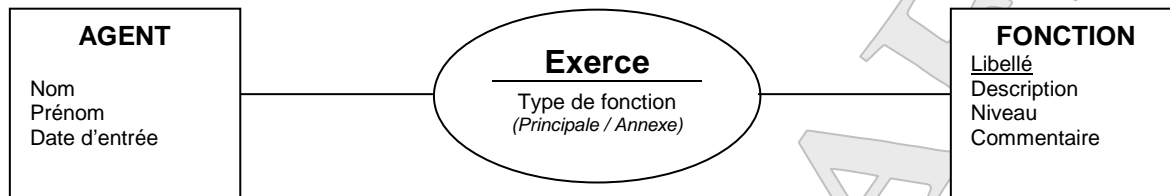


- A noter l'aspect grammatical du MCD : « l'agent » occupe « le bureau ». Les entités jouent le rôle de sujet ou de complément d'objet direct. L'association joue le rôle de verbe.

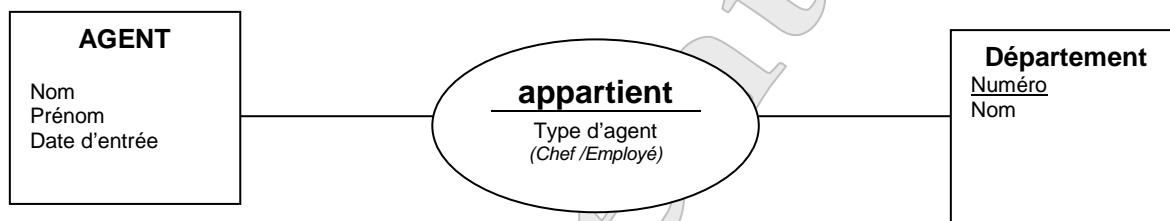
Etape 4 : Les relations porteuses de propriétés

- Il reste à traiter les données qui ne peuvent être affectées à une entité en particulier : celles qui dépendent de plusieurs entités ou qui évoluent dans le temps. Ces données sont des compléments circonstanciels qui qualifient l'action (où, quand...)

Exemple : le type de fonction « Principale / Annexe » dépend à la fois de la fonction et de celui qui l'exerce. Cette propriété n'a de sens que s'il existe une fonction correspondante et un agent pour l'exercer à titre principal ou non.



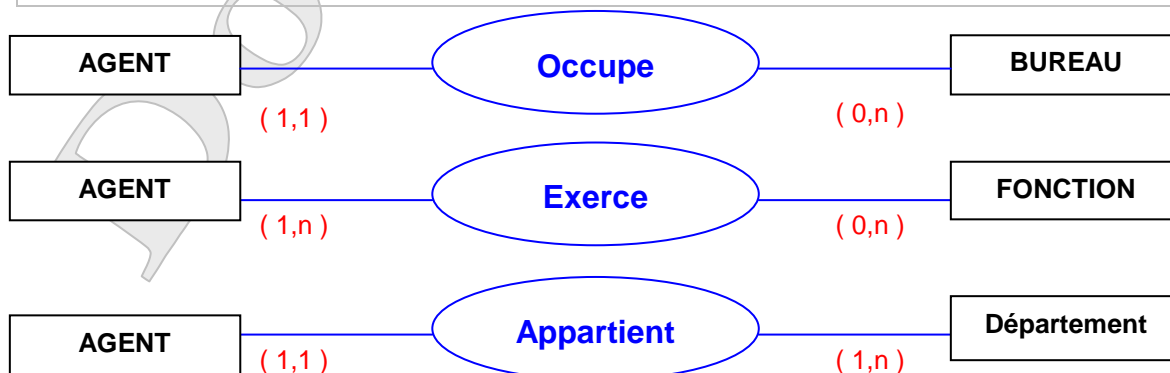
- De même, tous les agents appartenant à un département donné n'en sont pas chef. Cette qualité n'est propre ni à l'entité « Agent » ni à l'entité « Département ». Elle est donc propriété d'une relation qui peut s'établir comme suit :



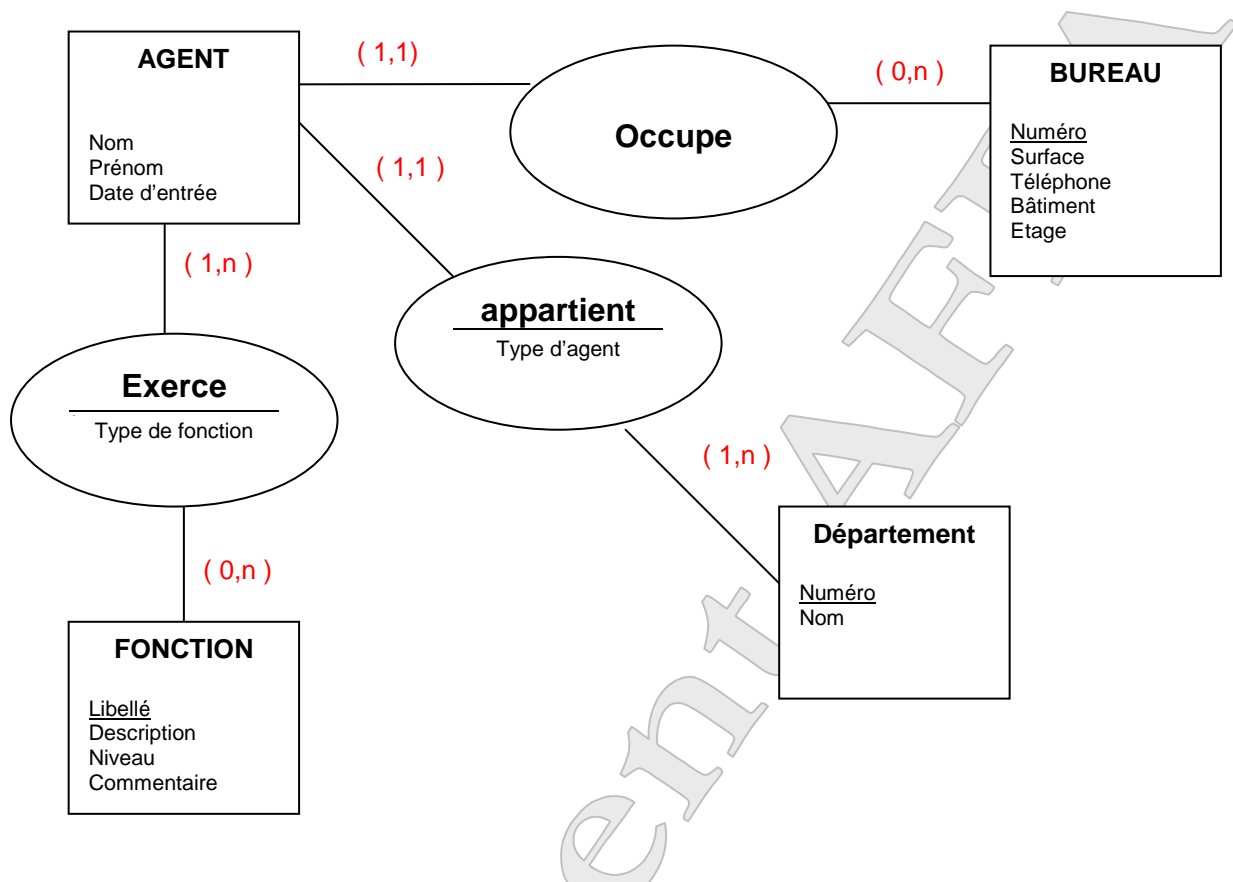
Etape 5 : cardinalités

On représente ici des règles de gestion exprimant le fait que la relation peut exister une fois, plusieurs fois, ou ne pas exister dans certains cas.

1 AGENT	occupe	1 BUREAU	Cardinalités (1,1)
1 AGENT	exerce	1 à 2 FONCTIONS	Cardinalités (1,n)
1 AGENT	appartient à	1 Département	Cardinalités (1,1)
1 BUREAU	peut être occupé par	0 à n AGENT	Cardinalités (0,n)
1 FONCTION	peut être exercée par	0 à n AGENT	Cardinalités (0,n)
1 Département	emploie	1 à n AGENT	Cardinalités (1,n)



Etape 6 : MCD final



Syndicat GROUPEDI

D'après le descriptif ci-dessous, établir le modèle conceptuel des données, pour informatiser la gestion de ce syndicat.

Le syndicat GROUPEDI est une association d'éditeurs, dont l'objectif est de fournir à ses membres des informations sur :

- Les auteurs.
- Les livres.
- Les magasins et les ventes.
- Les éditeurs.

Voici quelques phrases recueillies pendant les premiers interviews avec le personnel et la direction :

- Les auteurs écrivent des livres. Un auteur est identifié par un code à 9 chiffres ; il possède un nom, un prénom, une adresse et un téléphone.
- Chaque livre est identifié par un code à 6 caractères ; il possède également un titre, un type et un prix unitaire.
- Chaque auteur peut écrire plusieurs livres. Un livre peut parfois être corédigé par plusieurs auteurs.

- *Les types de livres sont codifiés par un mot clef.*
- *Un livre ne peut être publié que chez un seul éditeur.*
- *Les éditeurs emploient du personnel. Chaque employé est identifié par un code alphanumérique unique. Il possède bien-sûr un nom et un prénom. Il travaille chez un et un seul éditeur. On ne souhaite pas conserver l'historique de la carrière de chaque employé.*
- *Divers types d'emploi ont été repérés. Chaque employé exerce un et un seul type d'emploi.*
- *Les magasins passent des commandes pour s'approvisionner. Chaque commande peut concerner plusieurs livres. Chaque magasin a son propre système de numérotation des commandes (donc deux magasins peuvent très bien émettre le même numéro de commande !)*

Vos notes complémentaires, suite à la discussion avec le client :

-
-
-
-
-



Merise I, chap. II : « Exercices intermédiaires 1 et 2 » (corrigés en annexe)



Merise I, chap. II : « Exercice de synthèse 1 » (corrigé en annexe)

3. LE MODELE LOGIQUE DE DONNEES (MLD)



Merise I : Chap. VI, Modèle logique de données, mise en place

Le modèle logique des données est une représentation du modèle conceptuel en terme d'organisation des données, qui se réfère aux possibilités technologiques des SGBD. A ce stade de l'analyse, il faudra distinguer :

- Les bases de données hiérarchiques ou navigationnelles (de plus en plus rares)
- Les SGBD relationnels (l'état de l'art actuel, même si les bases de données objet progressent petit à petit)



Merise I, chap. VI : quelques notions sur les SGBD hiérarchique et réseau (« I Hiérarchie à un niveau », « II Hiérarchie à plusieurs niveaux », « III Réseau »

3.1 Plan du passage du M.C.D au M.L.D

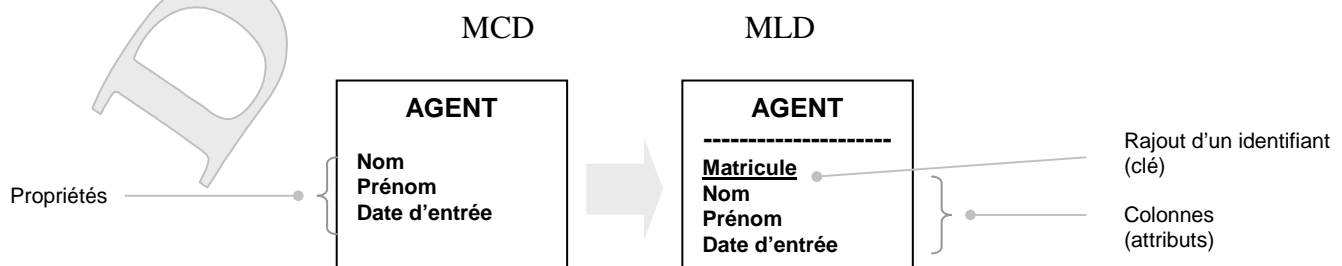
A partir du M.C.D on obtient un M.L.D « brut » grâce aux règles de passage :

- Etape 1 : trouver un identifiant pour chaque entité
- Etape 2 : les entités deviennent des tables.
- Etape 3 : traduire les relations du M.C.D, en distinguant les relations de cardinalité « 1-n » et « n-n »

3.2 Etape 1 : trouver un identifiant pour chaque entité

- Trouver un identifiant pour chaque entité qui assure l'unicité de ses occurrences.
- S'il n'en existe pas, **le rajouter** (type numéro ou alphanumérique, voire concaténation de deux identifiants liés ou plus).

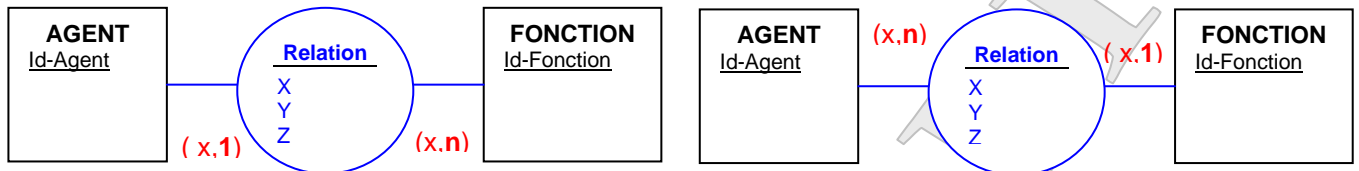
3.3 Etape 2 : les entités deviennent des tables



3.4 Traduire les relations du MCD

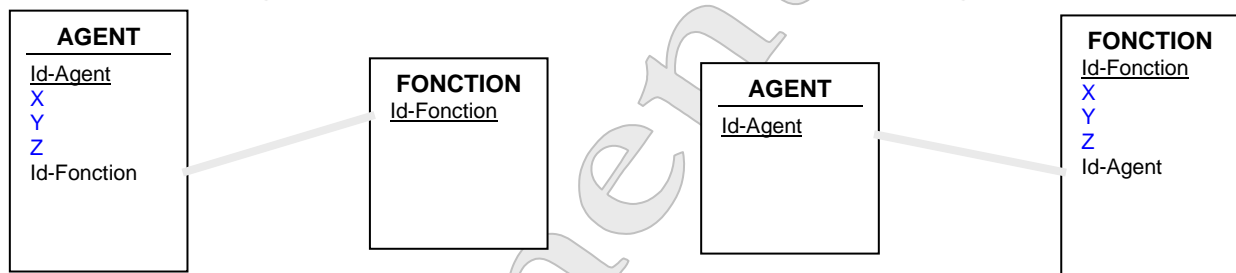
Cas d'une relation «1-n» ou «n-1»

Pour une relation donnée du M.C.D, on considère ses cardinalités maxi :

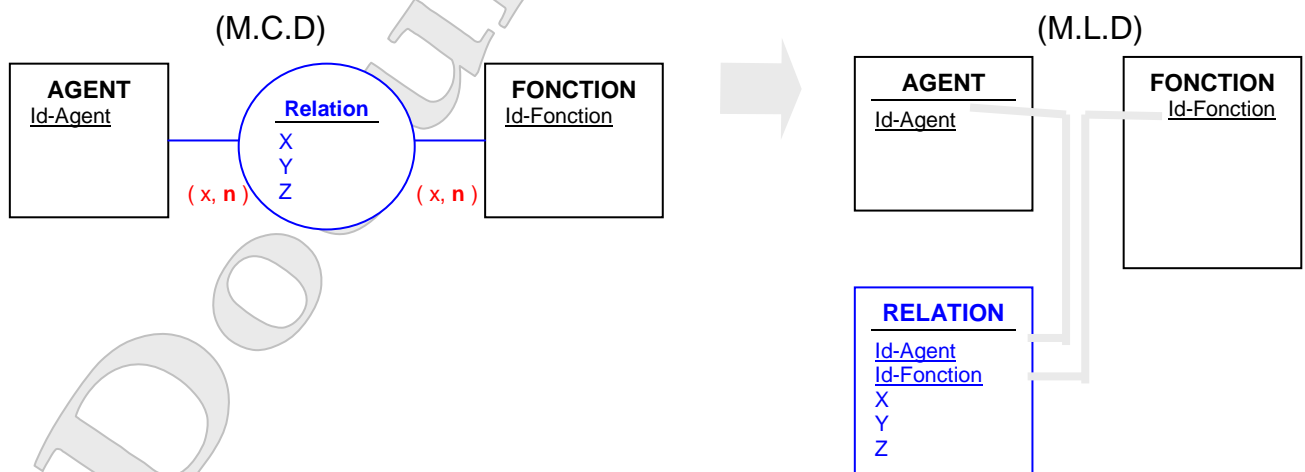


L'identifiant côté « n » devient clé étrangère de la table côté « 1 » et les propriétés de la relation glissent dans la table côté « 1 »

(M.L.D)

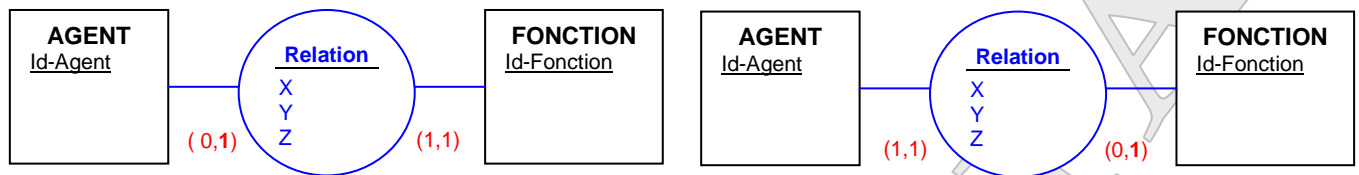


Cas d'une relation « n-n »

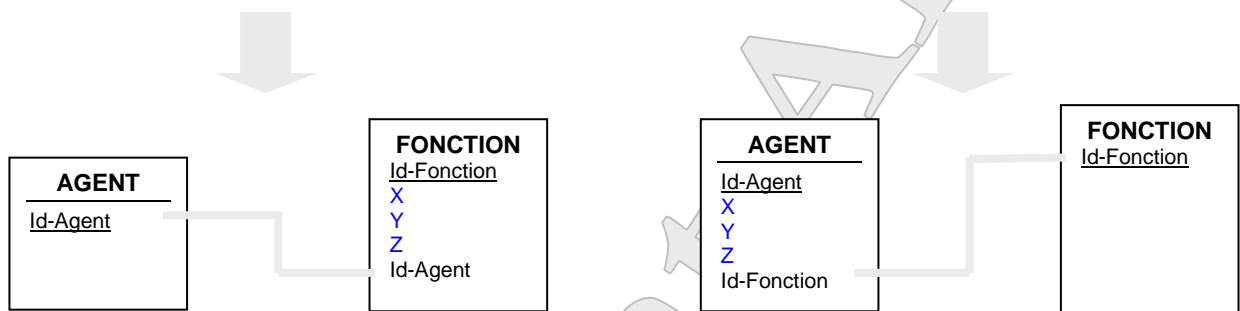


Toute relation de type « n-n » devient une table dont la clé primaire est la concaténation des identifiants des entités qu'elle liait entre elles. Les deux identifiants importés dans la nouvelle table deviennent des clés étrangères, vers les deux entités

Cas d'une relation « 1-1 »



L'identifiant côté (0,1) devient clé étrangère de la table côté (1,1) et les propriétés de la relation glissent dans la table côté (1,1)



Syndicat GROUPEDI (passage au MLD)

Construire le Modèle Logique des Données, en appliquant les règles de passage. Comparer avec la base PUBLI utilisée pour les exercices SQL.

N.B: Le descriptif a été simplifié pour une première initiation. Des tables de la base PUBLI pourraient donc ne pas être retrouvées. De même tous les attributs ne sont pas évoqués. L'essentiel dans cet exercice est de retrouver l'ossature de la base : Tables, clefs primaires, clefs étrangères.

4. CAS PARTICULIERS DE RELATION

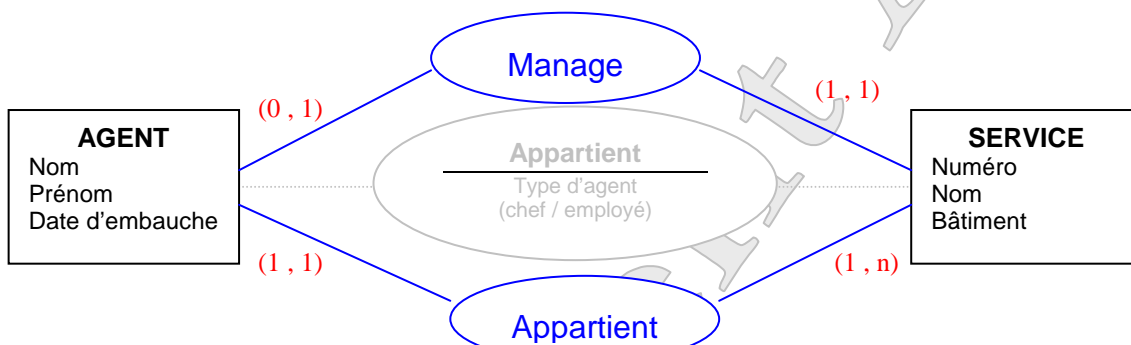
4.1 Relation à rôle



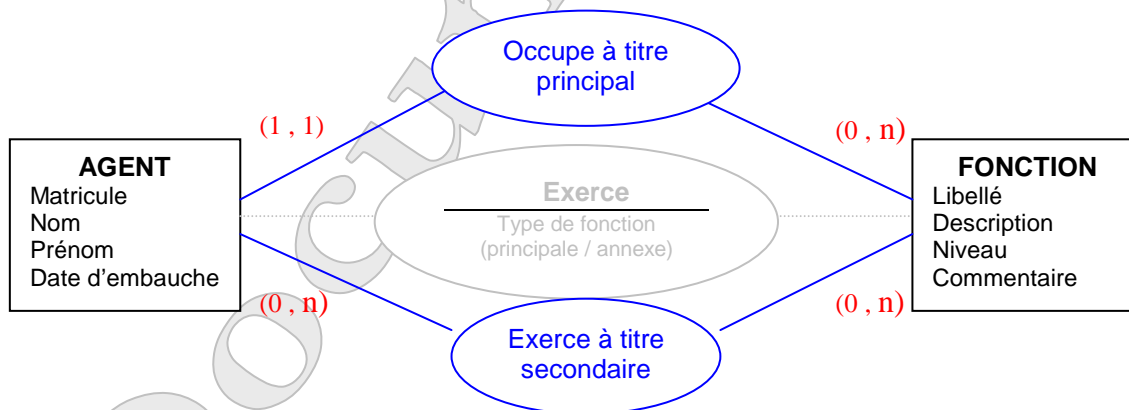
Merise 1 : Chap. IV, relations à rôle

Type d'agent : chef de service ou employé ?

Modélisons le fait qu'un agent peut être le chef du service ou un employé :



Type de fonction : principale ou annexe ?





Entreprise BISUS (MCD et MLD)

L'entreprise BISUS est une société de service en informatique. Elle travaille au forfait sur des projets qui lui sont confiés par des clients.

L'entreprise BISUS comporte plusieurs départements. Chaque département comporte des employés et un chef.

Chaque employé peut travailler sur un ou plusieurs projets

Réponses complémentaires du client :

-
-
-



Merise I, chap. IV : Exercices « Relations à rôle 1 à 3 » (corrigés en annexe)

4.2 Relation n-aires

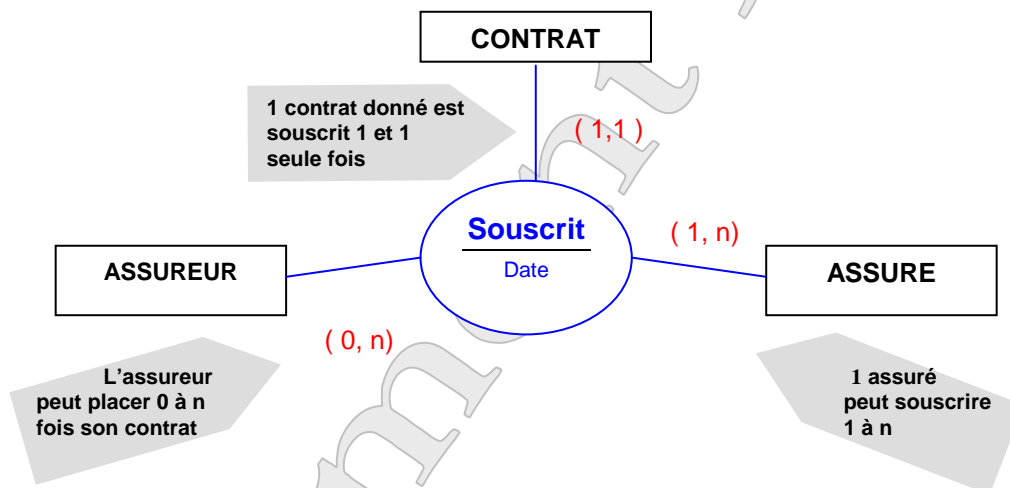


Merise I : Chap. III, cas particuliers de relation

Outre les relations binaires, il existe des relations ternaires (souvent), quaternaires (parfois), quinternaires et au-delà (excessivement rares).

Relation ternaire

- L'exemple ci-dessous établit une relation ternaire puisqu'elle dépend de 3 entités à la fois. En effet, la signature d'un contrat d'assurance met en présence 1e **CONTRAT**, l'**ASSUREUR** et l'**ASSURE**. Si une seule de ces 3 entités est manquante, la relation n'a plus de sens ou plus de raison d'être.



- Les cardinalités consistent à dénombrer le minimum et le maximum d'occurrences possibles de la relation, vue depuis une occurrence de chacune des entités.

Les C.I.F (Contrainte d'Intégrité Fonctionnelle)



Merise I : Chap. III, « Contraintes d'intégrité fonctionnelle »

- Les relations *n-aires* sont difficiles à gérer ensuite dans les modèles logiques et physiques.
- Il faut donc les simplifier à chaque fois qu'il est possible grâce aux C.I.F.
- Il y a C.I.F dès qu'une dépendance fonctionnelle apparaît entre 2 entités (relation obligatoire et unique).

Recherche des CIF dans la relation « Souscrit »

- Cas ASSUREUR / CONTRAT :

Connaissant l'ASSUREUR on ne connaît que les CONTRATS dont il est responsable. Ce n'est pas une C.I.F.

- Cas CONTRAT / ASSUREUR :

Connaissant 1 CONTRAT, on connaît forcément le seul ASSUREUR à en être responsable.

C'est une C.I.F

- Cas CONTRAT / ASSURE :

Connaissant 1 CONTRAT, on connaît forcément le seul ASSURE qu'il concerne.

C'est une C.I.F

- Cas ASSURE / CONTRAT :

Connaissant 1 ASSURE, on ne connaît que le ou les CONTRATS qu'il a souscrit(s). Ce n'est pas une C.I.F.

- Cas ASSUREUR / ASSURE :

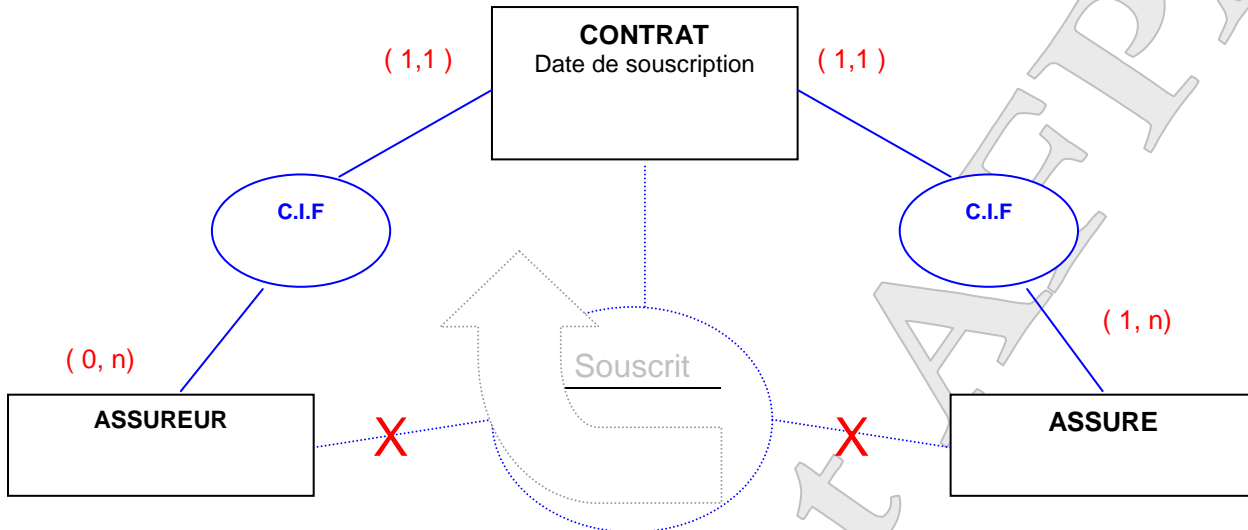
Connaissant 1 ASSUREUR, on ne connaît que les ASSURES dont il est responsable. Ce n'est pas une C.I.F.

- Cas ASSURE / ASSUREUR :

Connaissant un ASSURE on ne connaît que les ASSUREURS avec qui il a souscrit. Ce n'est pas une C.I.F.

Réductibilité des relations n-aires par les C.I.F

Réécriture de la précédente relation ternaire :



Pour réduire une relation n-aire, toujours éliminer sa branche opposée à celle de cardinalité $(1,1)$ de la C.I.F



Le « Cargo Maison de la culture » (MCD et MLD)

Cahier des charges

La maison de la culture souhaite informatiser son système de réservation.

Elle programme en début d'année un certain nombre de spectacles. Etablir le modèle conceptuel et le modèle logique d'un système capable de mémoriser les réservations et d'empêcher de réserver plusieurs fois la même place un même jour.

Synthèse du premier entretien avec le client

- *La maison de la culture possède trois salles : petite salle, grande salle, théâtre mobile.*
- *Dans chaque salle, les rangées de fauteuils sont numérotées de A à Z. Dans chaque rangée, les fauteuils sont numérotés de 0 à N.*
- *Une place du Cargo est donc référencée de façon unique par le triplet : nom de la salle, lettre de la rangée dans la salle, numéro de siège dans la rangée.*
- *Programmation des spectacles : la mise au point d'un spectacle coûte cher ! On donne en général plusieurs « représentations » du même spectacle : une représentation est définie par le nom du spectacle et sa date (« Les misérables », le 3 juin 2001).*
- *Les salles sont spécialisées : un spectacle exigeant un décor particulier qui ne tient que dans une des trois salles, le spectacle aura toujours lieu dans la même salle.*
- *Règle de gestion : on ne donnera jamais plusieurs fois le même spectacle le même jour (« Les misérables » le 3 juin 2001, en matinée et en soirée)*
- *Point capital : le système doit garantir que la même place ne soit pas attribuée à plusieurs individus, pour une même représentation !*



Merise I, chap. III : « Exercice de synthèse 3 et 4 » (corrigés en annexe)



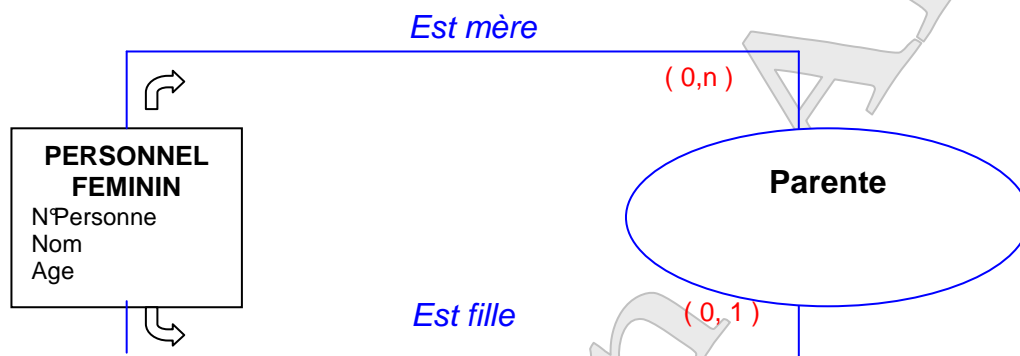
4.3 Relation réflexive



Merise I : Chap. IV, relation réflexive

Personnels apparentés

- Soit une relation portant sur une seule entité. Dans l'exemple ci-dessous, on souhaite gérer un personnel féminin en faisant apparaître les liens parentaux « mère de » et « fille de »
- Il s'agit d'une **relation réflexive** de l'objet « **PERSONNEL FEMININ** » sur lui-même.



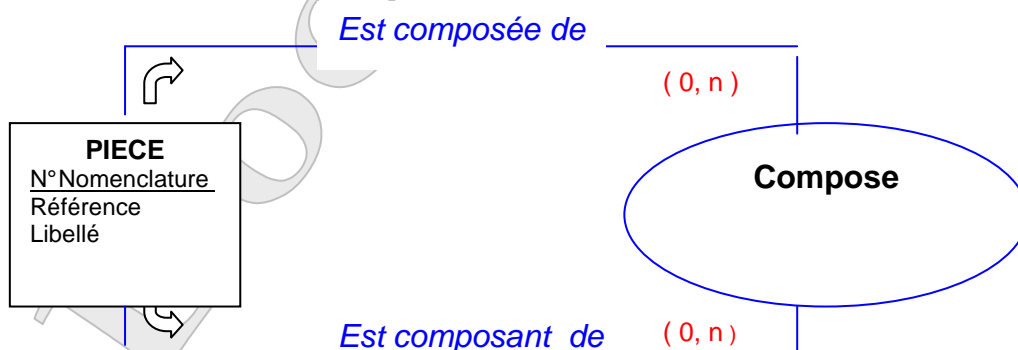
Cardinalités :

1 **PERSONNE** donnée peut « *être fille* » de 0 à 1 **PERSONNE** mère (qui ne travaille pas nécessairement dans cette entreprise !)

1 **PERSONNE** donnée peut « *être mère* » de 0 à plusieurs **PERSONNES** fille(s)

Modélisation d'une nomenclature

On veut gérer des pièces et ces mêmes pièces sont composantes d'autres pièces.



Cardinalités :

1 **PIECE** donnée peut « *être composant* » de 0 à plusieurs autres **PIECES**

1 **PIECE** donnée peut « *être composée* » de 0 à plusieurs autres **PIECES**



« **Firme YAKA** » (MCD et MLD)

Cahier des charges

1^{ère} partie

La firme YAKA assemble plusieurs modèles de micro-ordinateurs de type PC.

L'objectif de la base de données à construire est de mémoriser la nomenclature des produits manipulés par la firme YAKA.

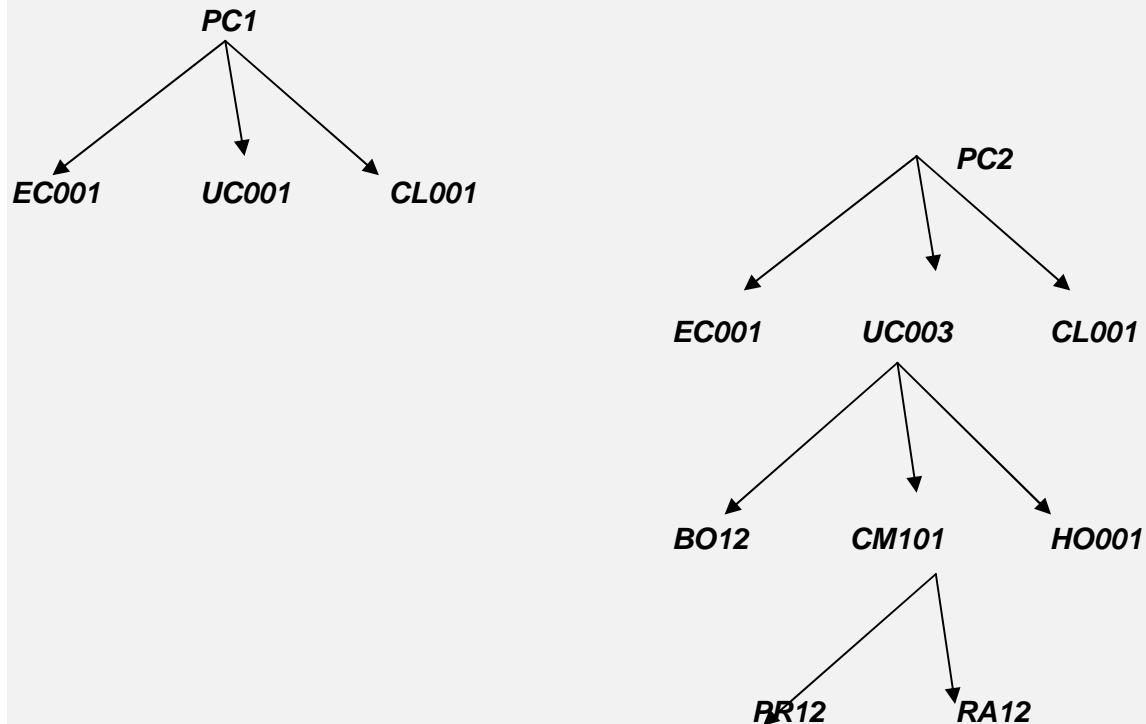
- *Les PC sont des assemblages de sous-ensembles, eux-mêmes formés de pièces élémentaires ou de sous-ensembles plus petits.*
- *PC, sous-ensembles et pièces élémentaires doivent être décrits dans la nomenclature.*
- *Une pièce ou un sous-ensemble peut intervenir dans plusieurs sous-ensembles ou produit. Un produit ou un sous-ensemble peut contenir plusieurs pièces ou sous-ensembles identiques.*
- *Les produits, sous-ensembles et pièces sont regroupés en classes : disques, écrans, chips etc...*
- *Chaque département de la société comporte des employés habilités à assembler certains sous-ensembles ou modèles de PC.*
- *Chaque département possède un chef.*

2^{ème} partie :

Compléter les modèles pour introduire une gestion commerciale. Il faut mémoriser les clients et les commandes.

Synthèse du premier entretien avec le client

- Chaque PC se compose, entre autres sous-ensembles, d'un Ecran (EC), d'une Unité Centrale (UC) et d'un clavier (CL).
- Chaque UC se compose d'un Boîtier (BO), d'une carte mère (CM) et d'une horloge (HO).
- La carte mère comprend un Processeur (PR) et de la mémoire RAM (RA)



Remarques :

- Certains employés sont habilités individuellement à assembler des cartes mères, d'autres des boîtiers, d'autres des PC entiers...
- La notion de « Classe » est indépendante de la nomenclature : c'est une typologie qui permet de regrouper toutes les cartes mères, tous les boîtiers..., indépendamment de leur utilisation dans l'assemblage des PC.

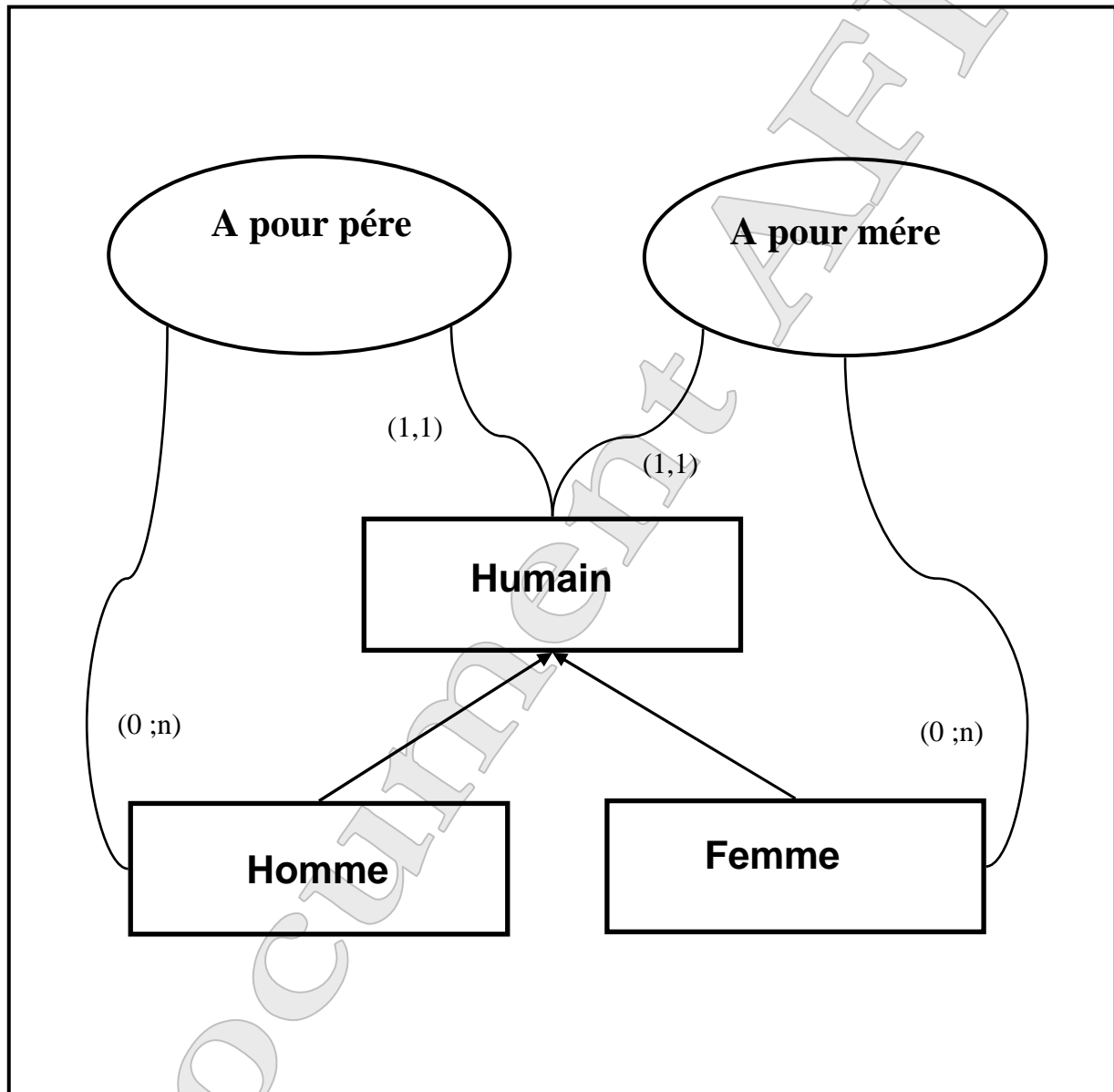


Merise I, chap. VI : « Exercice de synthèse 7 » (corrigés en annexe)

4.4 Relations et héritage.

L'exercice sur les mères et les filles, laisse de coté la moitié de l'humanité.

Pour une représentation correcte des filiations, il faut introduire une entité Humain et deux entités dérivées Homme et Femme. Ensuite il est facile de créer les associations « à pour père » et « à pour mère »



Passage au MLD, il y a trois possibilités :

- Trois tables (Humain, Homme, Femme)
- Une seule table (Humain)
- Deux tables (Homme et Femme)

Discuter.

Table des matières

.....	Erreur ! Signet non défini.
1. Présentation de la methode merise	3
1.1 Ressources	3
1.2 Comment se situer dans la méthode Merise	3
1.3 Mise en pratique de Merise dans ce document.....	5
1.4 Présentation succincte des différents modèles Merise.....	5
2. Le modèle conceptuel de donnees (MCD)	7
3. Le modèle Logique de donnees (MLD).....	12
3.1 Plan du passage du M.C.D au M.L.D	12
3.2 Etape 1 : trouver un identifiant pour chaque entité	12
3.3 Etape 2 : les entités deviennent des tables.....	12
3.4 Traduire les relations du MCD.....	13
4. Cas particuliers de relation	15
4.1 Relation à rôle.....	15
1.2 Relation n-aires.....	17
1.3 Relation réflexive.....	21