



UML

Passage du MCD au MLD

Rédigé par

David ROUMANET
Professeur BTS SIO

Changement

Date	Révision
Novembre 2020	Création
2022-09-17	Ajout des tables

Sommaire

A Règles de passage du MCD au MLD.....	1
A.1 Problématique.....	1
A.2 Le modèle relationnel.....	1
A.3 Objectifs du cours.....	2
B Règles de transformation.....	3
B.1 Relation binaire $X..1 - X..1$	3
B.2 Relation binaire $X..* - X..*$	4
B.3 Cas particuliers.....	6
B.3.1 Cas $1 - 1$	6
C Annexes.....	8
C.1 Sources complémentaires.....	8

A Règles de passage du MCD au MLD

A.1 Problématique

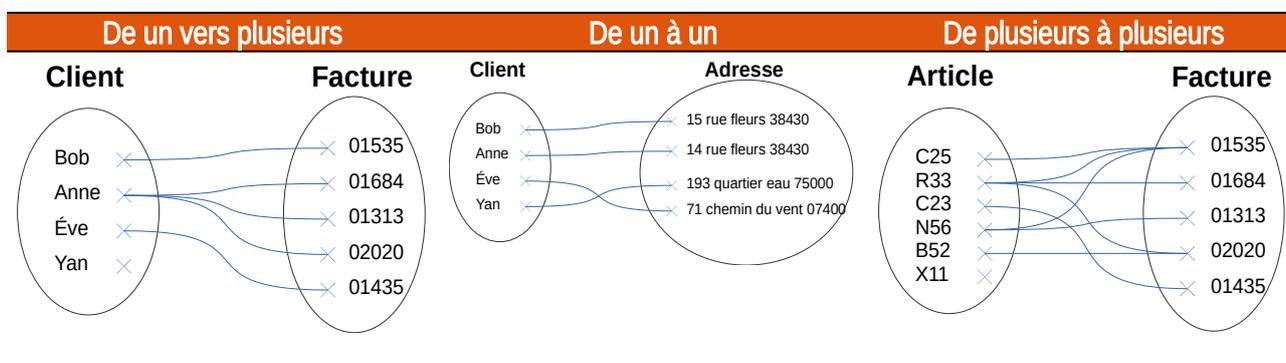
Le MCD est un modèle conceptuel : il doit s'affranchir des détails techniques.

Il permet de créer un modèle lisible et universel mais en étant abstrait, nécessite une phase d'adaptation pour permettre la création d'une base de données relationnelle. Le MLD est un modèle logique qui prend en compte les aspects pratiques des bases de données.

Pourquoi ne pas créer directement un MLD ? Quelles sont les règles pour passer du MCD au MLD ?

A.2 Le modèle relationnel

En modélisation de relation, nous avons les liens suivants :



Dans une base de données, la relation plusieurs vers plusieurs n'est pas directement possible pour un seul enregistrement. On ne peut pas utiliser de collection : les relations de plusieurs à plusieurs sont donc un problème pour retrouver facilement une information exploitable, et il est nécessaire de parcourir toute la liste des objets pour vérifier qu'un élément du premier groupe est utilisé dans le deuxième groupe.

En programmation, il suffirait de mémoriser une liste dans chaque élément, mais ce n'est pas faisable dans un SGBD. En SGBD, il y aura autant d'élément que de relation.

Ainsi, lors du passage du modèle conceptuel, il faudra discuter avec le client, pour ne pas nuire aux performances de la base, tout en fournissant une information utile et choisir une conception logique applicable.

A.3 Objectifs du cours

Les objectifs de ce cours – classés selon la taxonomie de Bloom – sont :

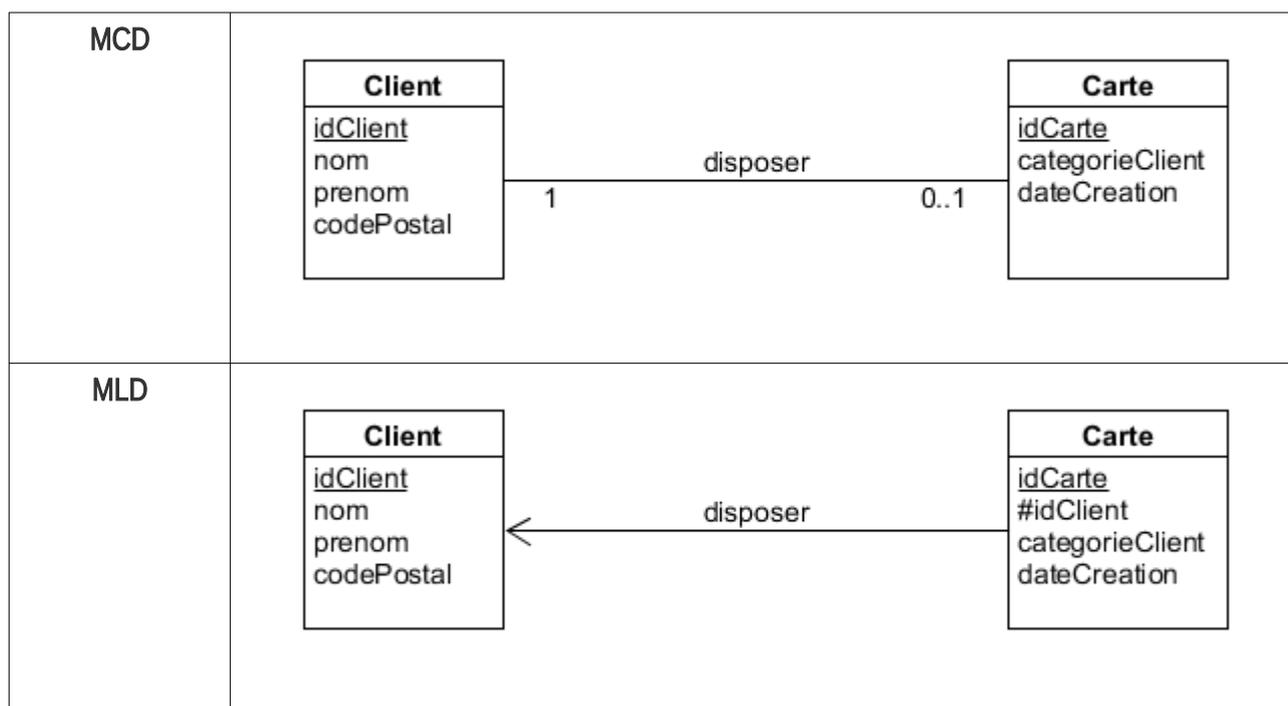
Mémoriser	Connaître les règles de conversion
Comprendre	Choisir les règles
Appliquer	Construire un modèle pratique et cohérent
Analyser	Savoir contourner les contraintes techniques des SGBD
Évaluer	–
Créer	–

B Règles de transformation

Voici les règles de bases pour convertir un MCD vers un MLD.

B.1 Relation binaire X..1 – X..1

Cette relation existe dans le cas d'un client qui peut avoir une carte de fidélité (mais il n'y est pas obligé).



Il suffit de placer la clé primaire de la table client en clé étrangère de la table carte.

<u>idCarte</u>	categorieClient	dateCreation	#idClient
115551	25	2022-08-31	227
115552	19	2021-04-19	511
115553	3	2021-12-22	684

Désormais, on retrouve facilement le client à partir de sa carte. Trouver la carte d'un client nécessite un parcours sur la table carte, mais si le client n'existe pas, c'est qu'il n'a pas de carte.



Une erreur serait de mettre `idCarte` comme clé étrangère dans la table `Client` : une recherche renverrait plusieurs clients qui auraient une carte NULL ce qui n'est pas indexable et représente une perte d'efficacité. Il est donc recommandé d'affecter la clé étrangère du côté qui reçoit plusieurs éléments.

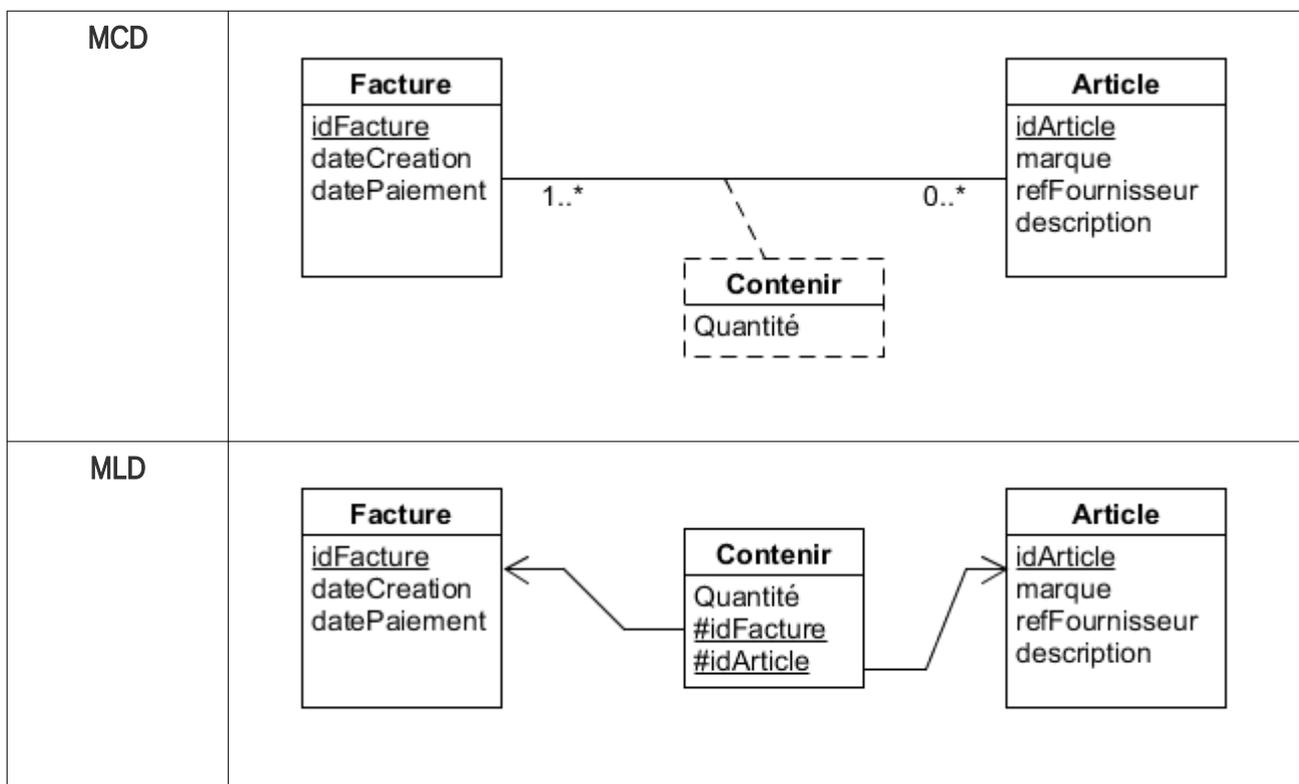
Dans le cas d'une relation 0..1 – 0..1 on pourra choisir l'une ou l'autre table, avec le risque cité précédemment.

La flèche indique donc, d'où provient la clé étrangère (elle débute généralement en face de la clé étrangère dans la table).

Ce schéma est celui qu'on trouve parfois dans PHPMyAdmin.

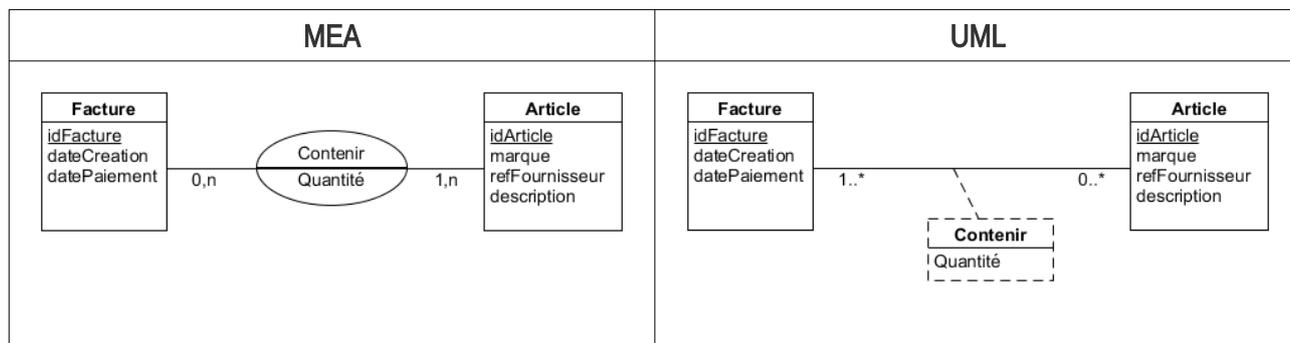
B.2 Relation binaire X..* – X..*

Dans ce cas de figure, on retrouve généralement des références d'articles dans un ensemble de factures : une facture peut contenir plusieurs types d'articles et un type d'articles peut être référencé dans plusieurs factures.



Dans cette relation, une classe-association indique la quantité d'un article sur une facture, mais même sans elle, dans le MLD il faut créer une table dont la clé primaire sera composée des clés étrangères des deux tables en relation. Ici, l'attribut quantité peut disparaître (la classe association du MCD disparaît d'abord, évidemment).

Une classe association remplace l'association dans un diagramme MEA (Merise).



Dans ce cas, il existera 3 tables ; Facture, Article et Contenir

Table Facture

idFacture	dateCreation	datePaiement
115551	2021-11-25	2021-11-25
115552	2021-11-25	2021-11-30
115553	2022-01-04	2022-02-02

Table Article

idArticle	marque	refFournisseur	Description
2010	25	SB435	Stylo noir mine 0.5mm
2011	19	SB438	Stylo rouge mine 0.7mm
2015	3	FB010	Feutre rouge

Table Contenir

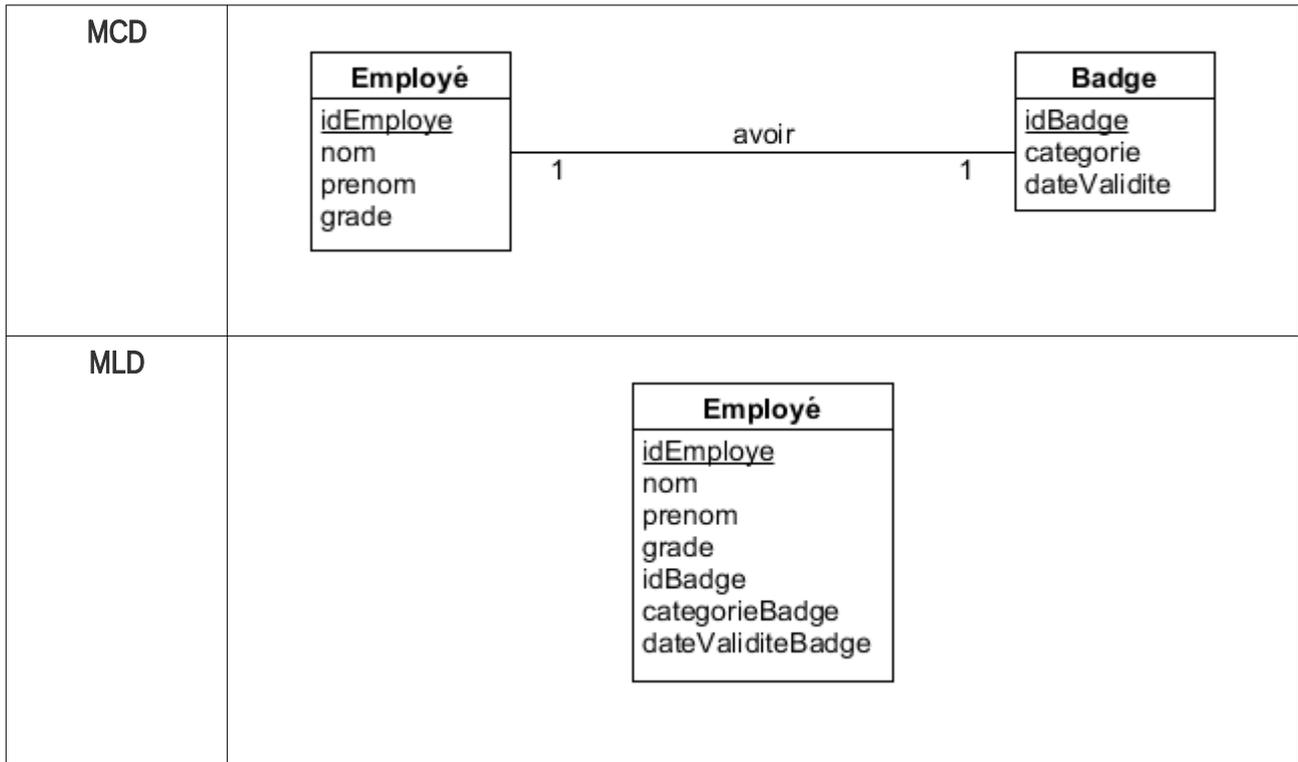
#idFacture	#idArticle	Quantite
115551	2010	1
115551	2015	1
115553	2011	5
115552	2010	10

La table Contenir permet de retrouver tous les articles d'une facture ou inversement, toutes les factures contenant un article précis.

B.3 Cas particuliers

B.3.1 Cas 1 – 1

Selon toute logique, si on associe un utilisateur à un badge, il est peut-être inutile de créer une table spécifique :



Cependant, il reste intéressant et pratique de conserver les deux tables séparées et – dans ce cas – placer en clé étrangère de chaque table, la clé primaire de l'autre. Cela reste une opération risquée, car il y a une redondance qui implique la mise à jour simultanée des deux tables :

- En affectant un badge à un employé, il faut écrire l'ID du badge dans la table Employé et simultanément écrire l'ID de l'employé dans la table badge.
- Si un employé perd son badge, que faut-il faire de l'enregistrement correspondant au badge perdu.

Table **Employe**

<u>idEmploye</u>	nom	...	<u>idBadge</u>	categorieBadge	dateValiditeBadge
2010	DUPONT		04016	admin	9999-12-31
2011	SANCHEZ		04023	employe	2023-06-07

2015	VAN LEYER		04099	visiteur	2022-05-14
------	-----------	--	-------	----------	------------

C Annexes

C.1 Sources complémentaires

Vidéos – Cours UML (Emds) : <https://www.youtube.com/watch?v=dJd6azZr9Kg&list=PLRR7wjtXb1cBQCE8ddM0B1D9DFj-WL3BX>

Vidéo – Cours MCD (Emds) : <https://www.youtube.com/watch?v=VFHVNA8xgK0>

Support web – Laurent Audibert : <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-classes>

Support web – interface versus classe abstraite : <https://waytolearnx.com/2019/04/difference-entre-une-interface-et-une-classe-abstraite-en-java.html>