



## Modèle conceptuel des données

### MERISE - Modèle conceptuel des données



#### Conception de bases de données

MERISE

Modèle conceptuel de la communication

Modèle conceptuel des données

Contraintes sur rôles

Contraintes sur relation

Modèle conceptuel des traitements

Modèle organisationnel des traitements

Modèle logique des données

Modèle physique

#### A lire aussi

Base de données

SQL

PL/SQL

LDAP

Active Directory

MySQL

Oracle

PostgreSQL

### Modèle conceptuel des données

Le modèle conceptuel des données (**MCD**) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités.

## Entités et classe d'entité

Une entité est la représentation d'un élément matériel ou immatériel ayant un rôle dans le système que l'on désire décrire.

On appelle **classe d'entité** un ensemble composé d'entités de même type, c'est-à-dire dont la définition est la même. Le classement des entités au sein d'une classe s'appelle *classification* (ou *abstraction*). Une entité est une *instanciation* de la classe. Chaque entité est composée de propriétés, données élémentaires permettant de la décrire.

Prenons par exemple une *Ford Fiesta*, une *Renault Laguna* et une *Peugeot 306*. Il s'agit de 3 entités faisant partie d'une classe d'entité que l'on pourrait appeler *voiture*. La *Ford Fiesta* est donc une instanciation de la classe *voiture*. Chaque entité peut posséder les propriétés *couleur*, *année* et *modèle*.

Les classes d'entités sont représentées par un rectangle. Ce rectangle est séparé en deux champs :

- le champ du haut contient le libellé. Ce libellé est généralement une abréviation pour une raison de simplification de l'écriture. Il s'agit par contre de vérifier qu'à chaque classe d'entité correspond un et un seul libellé, et réciproquement
- le champ du bas contient la liste des propriétés de la classe d'entité

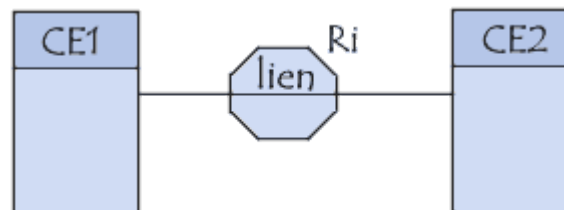


## Relations et classes de relation

Une relation (appelée aussi parfois *association*) représente les liens sémantiques qui peuvent exister entre plusieurs entités. Une classe de relation contient donc toutes les relations de même type (qui relient donc des entités appartenant à des mêmes classes d'entité). Une classe de relation peut lier plus de deux classes d'entité. Voici les dénominations des classes de relation selon le nombre d'intervenants :

- une classe de relation **récursive** (ou *réflexive*) relie la même classe d'entité
- une classe de relation **binaire** relie deux classes d'entité
- une classe de relation **ternaire** relie trois classes d'entité
- une classe de relation **n-aire** relie n classes d'entité

Les classes de relations sont représentées par des hexagones (parfois des ellipses) dont l'intitulé décrit le type de relation qui relie les classes d'entité (généralement un verbe). On définit pour chaque classe de relation un identificateur de la forme  $R_i$  permettant de désigner de façon unique la classe de relation à laquelle il est associé.

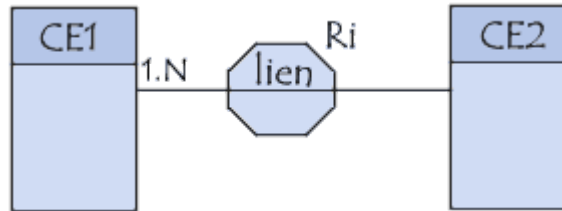


On peut éventuellement ajouter des propriétés aux classes de relation.

## La cardinalité

Les cardinalités permettent de caractériser le lien qui existe entre une entité et la relation à laquelle elle est reliée. La cardinalité d'une relation est composée d'un couple comportant une borne maximale et une borne minimale, intervalle dans lequel la cardinalité d'une entité peut prendre sa valeur :

- la borne minimale (généralement 0 ou 1) décrit le nombre minimum de fois qu'une entité peut participer à une relation
- la borne maximale (généralement 1 ou n) décrit le nombre maximum de fois qu'une entité peut participer à une relation



Une cardinalité 1.N signifie que chaque entité appartenant à une classe d'entité participe au moins une fois à la relation. Une cardinalité 0.N signifie que chaque entité appartenant à une classe d'entité ne participe pas forcément à la relation.

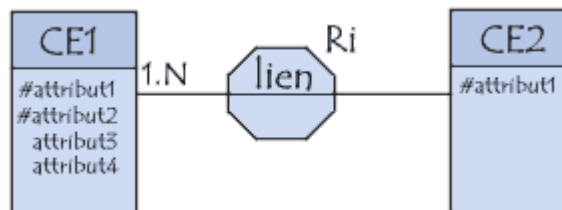
## Les identifiants

Un identifiant est un ensemble de propriétés (une ou plusieurs) permettant de désigner une et une seule entité. La définition originale est la suivante :

L'identifiant est une propriété particulière d'un objet telle qu'il n'existe pas deux occurrences de cet objet pour lesquelles cette propriété pourrait prendre une même valeur.

Les attributs d'une classe d'entité permettant de désigner de façon unique chaque instance de cette entité sont appelés *identifiants absolus*.

Le modèle conceptuel des données propose de faire précéder d'un # les identifiants (parfois de les souligner).



Ainsi, chaque classe d'entité doit posséder au moins un attribut identifiant, et l'ensemble de ses attributs identifiants doivent être renseignés à la création de l'entité.

## Agrégation (ou identification relative)

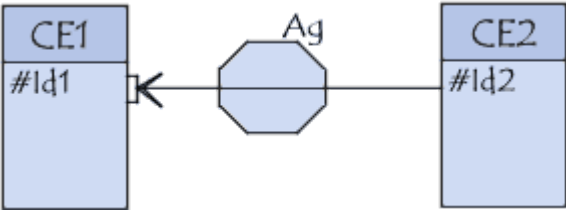
Lorsqu'un identifiant est constitué uniquement d'attributs intrinsèques à une entité, c'est-à-dire ne faisant référence à aucune autre entité, on le nomme **identifiant absolu**. Les entités comportant des identifiants absolus peuvent être définies indépendamment des autres occurrences d'entités, on dit que ces entités sont indépendantes.

Certaines entités ne peuvent toutefois être identifiées que par l'intermédiaire d'autres entités, c'est la raison pour laquelle on parle d'**identification relative**. On parlera par exemple de *la 4<sup>ème</sup> porte au 2<sup>ème</sup> étage du bâtiment B* au lieu de dire la porte n°3451...

Ainsi, l'**agrégation** (appelée aussi *identification relative*) permet de spécifier qu'une entité est nécessaire pour en identifier une autre.

- la classe d'entité permettant d'identifier est appelée *classe d'entité agrégante*
- la classe d'entité identifiée est appelée *classe d'entité agrégée*

La représentation de ce type de relation est la suivante :



**Discussions pertinentes trouvées dans le forum**

07/03 19h44	MCD gestion parc informatique	Développement	06/02 11h49->louati Mouna	11
02/10 22h27	Cherche prog qui fait des zolis MCD (MERISE)	Développement	18/11 07h35->Galsungen	24
18/06 13h40	Base de donnees (A partir MCD)	Développement	31/07 12h09->nabila	16
13/10 17h25	Help I need Help sur un exo de MCD Please	Développement	10/10 23h06->Cyolan31	15
29/07 13h46	MCD/Access	Développement	19/02 12h07->Doumbia	13
25/06 19h12	annales concours exos MCD MPD demande d aide	Développement	13/10 17h06->Gerard Toul	10
20/01 21h42	recherche outils MCD	Windows	16/04 22h05->sylvain	9
19/11 13h23	MCD d'un SGBD	Développement	09/02 10h49->Bel-hadji	7
31/10 00h28	le mcd de la gestion de stock en access	Logiciels/Pilotes	09/07 01h29->mounir	7
06/04 12h15	Modélisation (MCD)	Windows	06/04 16h54->Malika	6
Discussion fermée  Problème résolu		Plus de discussions sur « <b>MCD</b> »		

Ce document intitulé « MERISE - Modèle conceptuel des données » issu de l'encyclopédie informatique Comment Ça Marche ([www.commentcamarche.net](http://www.commentcamarche.net)) est mis à disposition sous les termes de la licence [Creative Commons](#). Vous pouvez copier, modifier des copies de cette page, dans les conditions fixées par la licence, tant que cette note apparaît clairement.

