

Annexe 12

Le modèle Entité-association de base

Compléments

Dans cette annexe, on revient sur l'interdiction d'utiliser certaines **contraintes de cardinalité**. On décrit le concept de **type d'entités faible**, classique dans certains modèles conceptuels et on illustre le danger d'utiliser des **identifiants primaires** porteur d'une **signification**. On y trouve aussi une collection d'exercices accompagnés, pour la plupart, d'une suggestion de solution.

A12.1 POURQUOI EXCLURE LES CARDINALITÉS [1-N] ?

<complément de la section 12.4.5>

Le modèle de base exclut certaines contraintes de cardinalité de rôle, notamment :

1. la cardinalité **[1-N]**,
2. la cardinalité **[1-1]** lorsque le rôle opposé est lui-même soumis à une cardinalité **[1-1]**.

Ces limitations peuvent être frustrantes lors de l'élaboration d'un schéma conceptuel. Ces valeurs permettraient en effet de modéliser plus précisément certaines structures du domaine d'application. Par exemple, il semblerait utile de préciser

- qu'une commande possède **au moins un** détail,
- qu'un client possède **au moins un** véhicule,

- qu'un contrat couvre **un** et un seul véhicule
- ou encore qu'un accident implique **au moins un** véhicule.

Pourquoi par conséquent ne pourrait-on pas accepter le schéma de la figure A12.1 ?

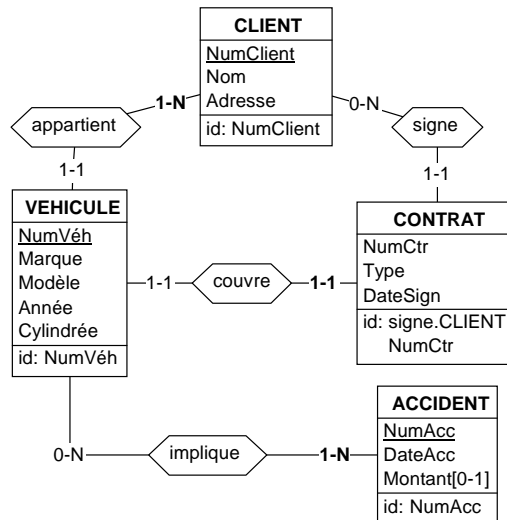


Figure A12.1 - L'usage des contraintes de cardinalité interdites

En réalité, cette interdiction ne concerne pas a priori le processus d'analyse conceptuelle proprement dit, mais bien celui de production du schéma de la base de données. La raison est simple : la traduction en SQL de ces cardinalités fait appel à des techniques complexes qui ne correspondent pas à l'objectif de simplicité de la démarche décrite. En revanche, nous les étudierons lors de l'étude de la démarche complète, pour laquelle ces valeurs de cardinalité seront admises.

Pour ne pas laisser le lecteur dans le brouillard, disons quelques mots sur l'origine de cette complexité, qui relève du paradoxe de *l'oeuf et de la poule*.¹ Lorsqu'on crée une entité VEHICULE, on l'associe simultanément à son entité CLIENT, sinon, il existerait pendant un certain temps, même très court, un véhicule sans propriétaire, ce qui est interdit par le schéma conceptuel. A l'inverse, lorsqu'on crée une entité CLIENT, il faut simultanément lui associer au moins une entité VEHICULE, sinon, il existerait pendant un certain temps, même très court, un client sans véhicule, ce qui est interdit par le schéma conceptuel.

Puisqu'aucune de ces deux entités ne peut être *officiellement* créée avant l'autre, il n'y a guère que deux solutions :

1. soit les créer simultanément via une restructuration du schéma (par exemple fusionner VEHICULE et CONTRAT) ou via une opération complexe (procédures SQL, déclencheurs *instead of*, méthodes de tables typées en SQL3, ORM),
2. créer l'une, puis l'autre dans la foulée, mais en demandant aux observateurs de

1. Lequel, de l'oeuf et de la poule, est apparu en premier ?

regarder ailleurs tant que la double opération n'est pas terminée (transaction). Nous attendrons, avant d'étudier ces techniques avancées, d'avoir acquis un peu d'expérience dans la construction des bases de données.

Outre ce raisonnement de nature technique, des raisons organisationnelles ou ergonomiques peuvent justifier qu'on traduise par la valeur **[0-N]** ce qui, en première analyse mériterait la valeur **[1-N]**. Reprenons le cas du type d'associations **appartient**, dont on vient de décider qu'il exprimait le fait que tout client possède un véhicule et que tout véhicule appartient à un client. Si cette modélisation est légitime du point de vue statique, elle ne l'est pas nécessairement du point de vue dynamique. S'il apparaît que dans la pratique on a l'habitude d'enregistrer un client alors même que le véhicule qu'il désire assurer n'est pas encore connu, alors il faut accepter que, pendant un certain temps, une entité CLIENT ne soit pas associée à une entité VEHICULE. Dès lors, on assignera au rôle appartient.CLIENT une cardinalité **[0-N]** plutôt que **[1-N]**. De même, bien qu'on admette que *toute commande doit spécifier au moins un produit*, le fait qu'on puisse enregistrer et vérifier une commande avant d'enregistrer ses détails peut nous inciter à choisir pour le rôle de.COMMANDE, la cardinalité **[0-N]** plutôt que **[1-N]**.

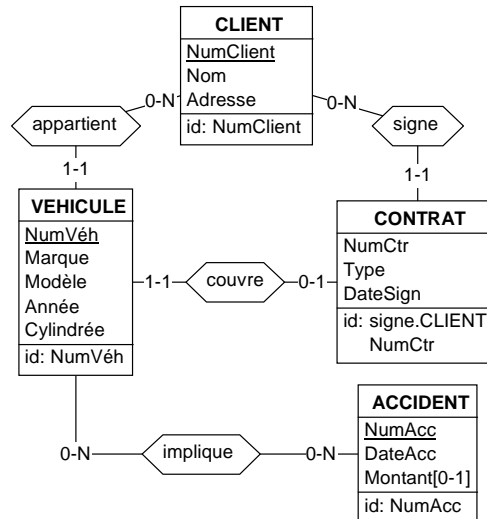
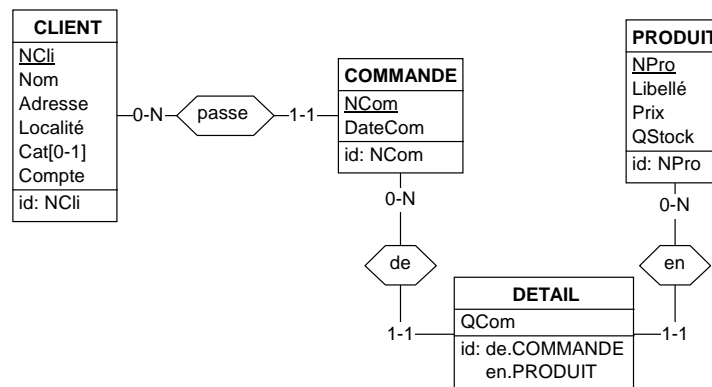
A12.2 TYPE D'ENTITÉS FAIBLE (WEAK ENTITY TYPE)

<complément de la section 12.6.1>

Un type d'entités est qualifié de *faible* lorsque l'existence de ses instances et leur identification dépendent de celles d'un autre type d'entités. Dans le schéma A12.2, CONTRAT est un *type d'entités faible* (dépendant de CLIENT) tandis que VEHICULE ne l'est pas.

Le concept de *weak entity type* a été introduit par Chen [Chen, 1976] et est repris, encore aujourd'hui, dans la plupart des modèles conceptuels nord-américains.

À première vue utile, il est malheureusement mal défini et irrégulier. Par exemple, le type d'entités DETAIL du schéma A12.3 pourrait être considéré comme **plus faible** que CONTRAT dans le schéma puisqu'il dépend, non pas d'un, mais de deux autres types d'entités. Cependant, il ne l'est pas puisque son identifiant (1) comporte deux rôles et (2) n'inclut pas d'attributs ! En raison de la présence de ce concept, ces modèles s'avèrent souvent déficients, même dans des situations très simples. Il est frappant de constater que ces modèles ne peuvent exprimer l'identifiant du type d'entités DETAIL. Le concepteur sera obligé d'introduire dans DETAIL l'attribut redondant NPro, formant avec COMMANDE l'identifiant de DETAIL, et assorti d'une contrainte d'intégrité référentielle de DETAIL vers PRODUIT, tout à fait incongrue dans un schéma conceptuel !

Figure A12.2 - CONTRAT est dit *faible*Figure A12.3 - DETAIL est un type d'entités deux fois dépendant mais qui n'obéit pas à la définition du type d'entités *faible*

L'ensemble des attributs d'un identifiant d'un identifiant hybride est parfois appelé identifiant *local* ou *relatif* de leur type d'entités. L'attribut NumCtr est un identifiant relatif de CONTRAT par rapport à CLIENT (via signe). Par contraste, un identifiant ne contenant que des attributs sera appelé *absolu*.

Une étude critique de ces concepts a été proposée dans la référence ci-dessous.

Référence

Mira Balaban, Peretz Shoval. *Resolving the "Weak Status" of Weak Entity Types in Entity-Relationships Schemas*. in Proc. Int. Conf. ER-1999, pp. 369-383, LNCS 1728 Springer, 1999

A12.3 IDENTIFIANTS PRIMAIRES : UNE ANECDOTE

<complément de la section 12.6.4>

Identifiant de désignation ou contrainte d'intégrité ? Un exemple réel va clarifier le propos. Dans une base de données répertoriant des établissements scolaires, on avait défini pour ceux-ci l'identifiant primaire comme suit :

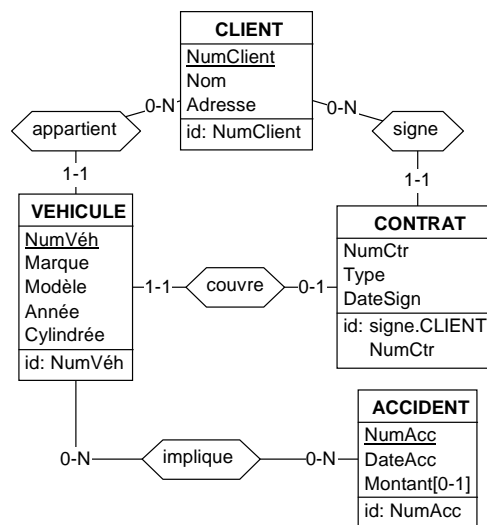
code postal de la commune de l'école + **initiales du nom** de l'école

On obtenait ainsi un code compact qui offrait en outre deux avantages : il était facile à reconstituer pour une école et il était possible d'en extraire la commune sans consulter les autres données. Ce choix qui semblait raisonnable au départ s'est avéré plus tard catastrophique. En effet, d'une part, il n'est pas rare qu'un établissement change de nom et donc d'initiales, et d'autre part, des regroupements de communes entraînent une modification du code postal de certains établissements. Après quelques années, le logiciel de gestion d'écoles utilisant cette base de données a dû être transformé suite à cette décision maladroite et en apparence anodine.

Cet exemple n'est pas un plaidoyer absolu en faveur des identifiants primaires de désignation, *non significatifs*, mais invite à se poser la question de la **stabilité** des valeurs des identifiants primaires lorsque les lois de comportement du domaine d'application sont susceptibles d'évoluer.

A12.4 EXERCICES DU CHAPITRE 12

A12.1 On considère le schéma ci-dessous. Admettons à présent que, toutes choses restant égales par ailleurs, (1) un véhicule puisse être couvert par un nombre quelconque de contrats, (2) on désire connaître, chaque fois qu'un véhicule est impliqué dans un accident, le pourcentage de responsabilité qui lui a été attribué. Modifier le schéma en conséquence.



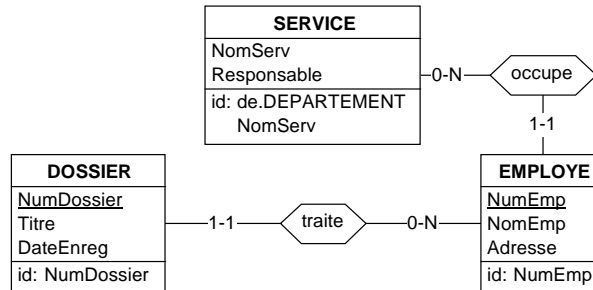
A12.2 En utilisant les conventions graphiques des figures 12.6 et 12.16, dessiner les populations (entités, associations, valeurs d'attributs) au format du schéma de la figure 12.19 et telles décrites par le contenu de la base de données de la figure 2.8.

A12.3 En utilisant le dessin élaboré lors de la résolution de l'exercice 12.2, retrouver les informations suivantes, selon le procédé de la section 12.8 :

- quels sont les produits commandés à Poitiers ?
- quels clients ont commandé le produit PA45 le 2/1/2016 ?

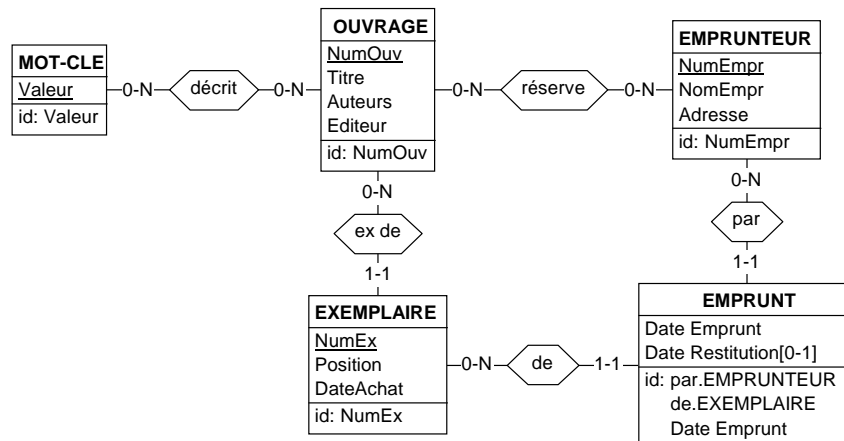
A12.4 Dans l'interprétation du schéma 12.25, on précise qu'un service est réputé traiter un dossier dès qu'un de ses employés est en charge de ce dossier. Modifier le schéma pour tenir compte de cette précision.

Solution



A12.5 Modifier le schéma de la figure 12.26 de manière à y représenter
 a) la réservation d'un ouvrage (ou d'un exemplaire ?) par un emprunteur;
 b) l'historique des emprunts d'ouvrages (ou d'exemplaires ?).

Solution



A12.6 Dessiner un graphe de populations représentatives pour le schéma de la figure 12.27. On y représentera une situation réelle, qu'on trouvera dans un horaire de chemin de fer.

A12.7 Pourquoi dans le schéma de la figure 12.27 un agent conduit-il un voyage et non un train, comme il est d'usage ?

A12.8 Le schéma de la figure 12.27 permet-il de résoudre le problème suivant ? Lorsqu'un conducteur X désire transmettre un colis à un autre conducteur Y, il dépose ce colis à une station par laquelle il passe, et par laquelle Y passera plus tard. X et Y se posent la question suivante à la date Z : où (quelle station) et à quelle date au plus tôt, Y pourra-t-il prendre livraison d'un colis que X

veut lui transmettre. On suppose qu'un voyage ne s'étend pas sur plus d'une journée, et qu'un colis déposé à une date déterminée pourra être enlevé dès le lendemain.

A12.9 Pourquoi dans le schéma de la figure 12.27 les sections ne sont-elles pas identifiées par leurs stations de départ et d'arrivée ?

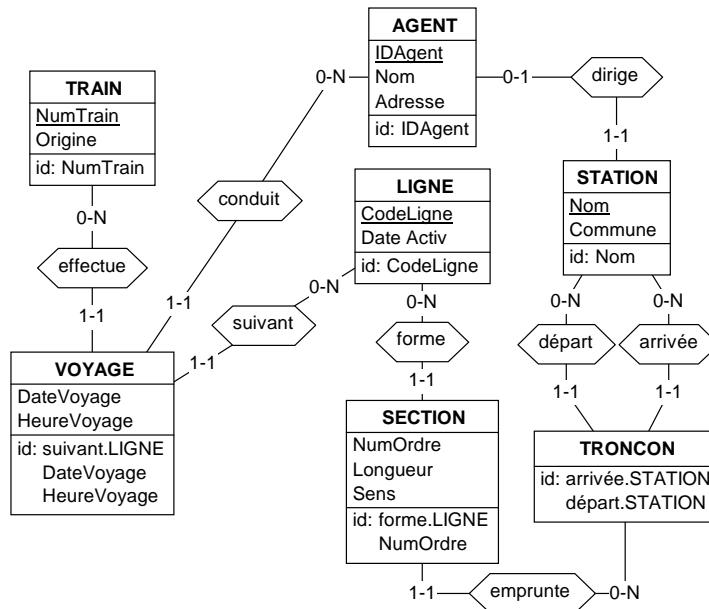
Solution

Parce qu'une section est un fragment de ligne et non la portion de voie qui relie deux stations. Il existe autant de sections entre deux stations qu'il y a de lignes passant consécutivement par ces stations.

A12.10 Le schéma de la figure 12.27 représente des sections de ligne de telle manière que deux lignes empruntant le même tronçon de voie définissent deux sections différentes. Donner au concept de *tronçon de voie* une définition précise et modifier le schéma conceptuel de manière qu'il représente non seulement les sections mais également les tronçons de voie.

Solution

Un **tronçon** est une liaison directe, ininterrompue, entre deux stations; il n'existe pas de station au milieu d'un tronçon. Conventionnellement, on fixe pour un tronçon une station de **départ** et une station d'**arrivée**. Il n'existe pas plus d'un tronçon entre deux stations, bien qu'un tronçon puisse comporter plusieurs portions de voies physiques (= rails). Une section d'une ligne emprunte un tronçon dans le *sens direct* (départ → arrivée) ou dans le *sens inverse* (arrivée → départ).



A12.11 Toujours relativement à ce même schéma, affiner le concept de *train*, en considérant que celui-ci est constitué de motrices, de voitures de voyageur, de wagons de marchandise, de fourgons, *etc.*, dans un ordre déterminé.

A12.12 On considère les deux types d'associations *r* entre A et B et *s* entre B et C. On considère aussi le type d'associations *rs* défini comme la composition de *r* et *s* (figure B.4). Déterminer la classe fonctionnelle de *rs* à partir de celles de *r* et de *s*. Cette question est très importante pour comprendre l'information contenue dans un schéma. En considérant le schéma de la figure 12.18, elle pourrait par exemple se concrétiser comme suit :

- combien de véhicules sont couverts par les contrats signés par un client ?
- combien de contrats couvrent les véhicules appartenant à un client ?
- combien de clients sont propriétaires des véhicules couverts par un contrat ?
- combien de propriétaires sont impliqués dans un accident ?
- combien de clients ont signé les contrats couvrant les véhicules d'un propriétaire (double composition) ?

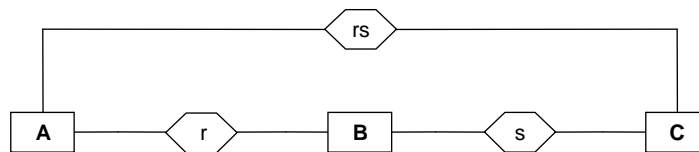


Figure A.4 - Etude de la composition de deux types d'associations

Solution

Pour simplifier le raisonnement, on considère les quatre classes fonctionnelles 1:N, N:1, 1:1 et N:N (N:1 est simplement la classe fonctionnelle d'un type d'association 1:N considéré dans le sens inverse). On définit ainsi 16 configurations distinctes.

r	s	rs
1:N	1:N	1:N
1:N	N:1	N:N
1:N	1:1	1:N
1:N	N:N	N:N
N:1	1:N	N:N
N:1	N:1	N:1
N:1	1:1	N:1
N:1	N:N	N:N
1:1	1:N	1:N
1:1	N:1	N:1
1:1	1:1	1:1
1:1	N:N	N:N
N:N	1:N	N:N
N:N	N:1	N:N

N:N	1:1	N:N
N:N	N:N	N:N