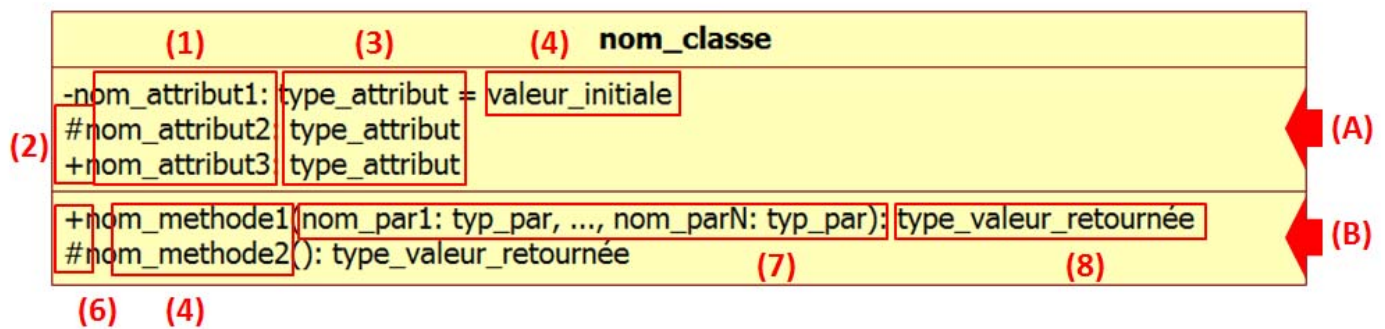


UML – Diagramme de classes de conception

Entités



Cette entité représente une *classe*. Chacune peut être décrite par un ou plusieurs *attributs* et proposer une ou plusieurs méthodes. Elle possède un nom explicite.

(A) C'est le compartiment de l'entité dans lequel sont listés les attributs.

(B) C'est le compartiment de l'entité dans lequel sont listés les méthodes.

(1) Le nom de l'attribut.

(2) La visibilité de l'attribut. En UML, on utilise les caractères +, - et #. (+ : *public*, - : *private*, # : *protected*).

(3) Le type de l'attribut.

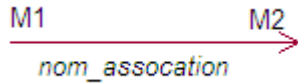
(4) La valeur initiale de l'attribut. Vous n'êtes pas obligé de mettre une valeur initiale à chaque attribut.

(5) Le nom de la méthode.

(6) La visibilité de la méthode. En UML, on utilise les caractères +, - et #. (+ : *public*, - : *private*, # : *protected*).

(7) La liste des paramètres de la méthode s'il y a des paramètres et le type de chaque paramètre.

(8) Le type de la valeur retournée par la méthode si la méthode retourne une valeur.



Cette entité représente une *association* entre deux *classes*. Elle peut avoir un nom mais cela n'est pas une obligation. Son nom doit permettre de qualifier l'*association* entre les deux classes. L'*association* est généralement orientée mais cela n'est pas une obligation. Enfin, chaque *association* dispose de deux indicateurs de *multiplicité* M1 et M2. Cela peut être un entier ou une expression telle que « 1...* » qui signifie « au moins 1 », ou * qui signifie « un nombre quelconque ».



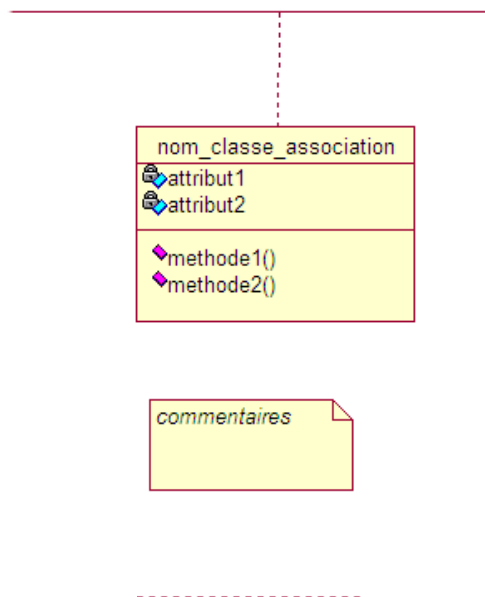
Cette entité représente une *relation de généralisation* entre classes conceptuelles et sous-classes conceptuelles ou entre classes conceptuelles et superclasses conceptuelles. Une classe est plus générale ou englobante qu'une sous-classe.



Cette entité représente une relation d'*agrégation composite*. Dans le texte 2.10, Craig Larman présente cette relation avec un losange plein. La multiplicité du côté du losange est toujours de 1.



Cette entité représente une relation d'*agrégation*, appelée aussi *agrégation partagée*. La différence avec la relation précédente est que la multiplicité du côté du losange peut être supérieure à 1.

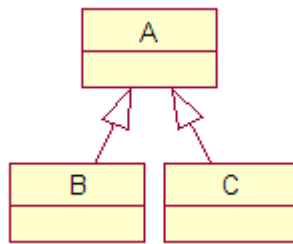


Cette entité représente une *classe d'association*. Comme toute classe, elle a un nom et possède un ensemble d'attributs et de méthodes qui caractérise les instances de la classe.

Cette entité permet d'inscrire un commentaire sur le diagramme de *classes conceptuelles*. Ce commentaire peut être relié à toute entité du diagramme.

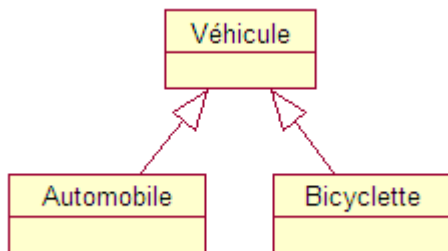
Cette relation permet de relier un commentaire à un élément du diagramme. Cet élément peut être une *classe conceptuelle* ou une *association*.

Sémantique élémentaire



La classe A est plus générale que les classes B et C.

Exemple



Un véhicule est une entité plus générale que les entités automobile et bicyclette. Toute automobile et toute bicyclette est un véhicule.



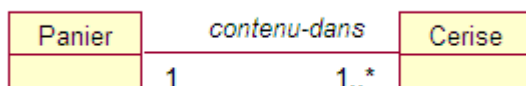
Une instance de la classe D est reliée à une ou plusieurs instances de la classe E.

Exemple 1



Un panier est relié à une ou plusieurs cerises.

Exemple 2



C'est un exemple proche du dernier. L'association a été nommée pour signifier plus clairement que les cerises sont contenues dans un panier.



N instances de la classe B sont en relation avec 1 instance de la classe composite A.

L'objet composite *main* est constitué de 5 *doigts*.

Exemple

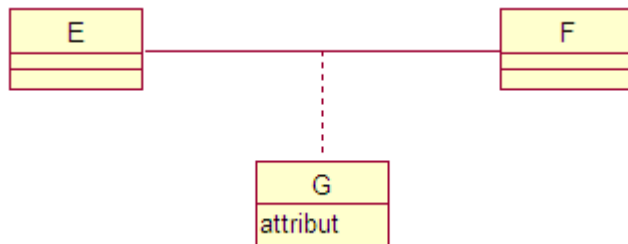


M instances de la classe D peuvent être contenues simultanément dans N instances de la classe C.

Exemple

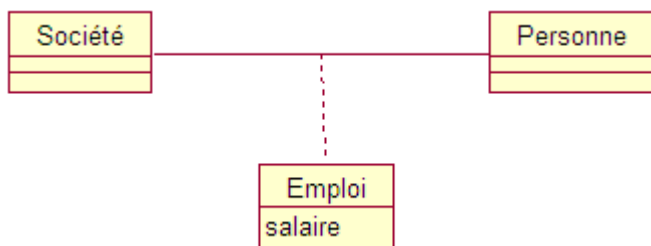


Il est très rare de voir ce type de relation dans le monde réel. Cela se retrouve plutôt dans le monde abstrait. L'exemple du texte 2.11 est repris ici : un nombre quelconque d'éléments UML peuvent se retrouver dans un nombre quelconque de paquetage UML.



L'instance de la classe E sont reliées aux instances de la classe F via la classe d'association G qui a un attribut *attribut*.

Exemple



Les instances des classes *Société* et *Personne* sont reliés par une association *emploi*. Or cette association a une certaine durée de vie et elle peut être caractérisée. De ce fait, il est préférable de créer une classe d'association *Emploi* dont l'un des attributs sera *salaire*.