

Diagrammes structurels d'UML 2



1. Diagramme de classes

Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. Ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classes et leurs relations.

1.1. Classe, attribut et opération

Une classe est un concept abstrait qui décrit un ensemble d'objets ayant les mêmes propriétés (attributs) et un même comportement (opérations). Une classe peut représenter des éléments variés comme :

- des éléments concrets (ex. : des avions) ;
- des éléments abstraits (ex. : service) ;
- des éléments d'une application (ex. : boîtes de dialogues) ;
- des éléments comportementaux (ex. : des tâches), etc.

1.1.1. Représentation UML d'une classe

Une classe se représente à l'aide d'un rectangle comportant plusieurs compartiments. Les trois compartiments de base sont :

- la désignation de la classe
- la description des attributs
- la description des opérations

Deux autres compartiments peuvent être aussi indiqués :

- la description des responsabilités qui énumère l'ensemble des tâches devant être assurées par la classe.
- la description des exceptions qui énumère les situations exceptionnelles devant être traitées et gérées par la classe.

Il est à noter qu'une description d'une classe peut être restreinte à un nombre réduit de compartiments selon les objectifs de la modélisation.

les symboles sont indiqués devant chaque attribut/opération pour signifier le niveau de visibilité autorisé pour les autres classes.

1.2. Association entre classes

Une association est une relation sémantique entre classes, qui décrit les connexions structurelles entre leurs instance. Une association est représenté par un lien reliant deux classes identifiée par un nom (forme verbale) et des multiplicités sur les extrémités. Il est possible d'exprimer ainsi des rôles sur le lien d'association.



Représentation UML d'une association

- Multiplicité d'association

La multiplicité spécifie le nombre d'instances d'une classe pouvant être liées à une seule instance d'une classe associée. le domaine de valeurs peut être décrit sous plusieurs formes : intervalle fermé, valeurs exactes, valeurs indéterminées. Ce présent tableau illustre les principales multiplicités :

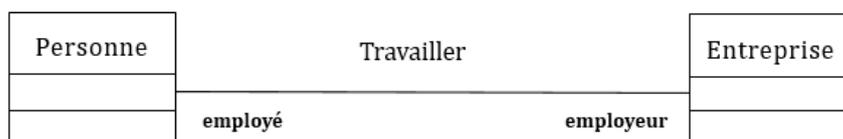
Domaine de valeurs	Interprétation
1	Une et une seule instance
0 .. 1	Zéro ou une instance
N	N instances (entier naturel)
M .. N	de M à N instances (entiers naturels)
*	De zéro à un nombre indéterminé d'instances
0 .. *	De zéro à un nombre indéterminé d'instances
1.. *	De un à un nombre indéterminé d'instances

Exemple : l'interprétation des multiplicités dans cet exemple est la suivante: à une instance de la classe A correspond 1 à nombre indéterminé d'instances de B et à une instance de la classe B correspond 2 à 10 instances de A.

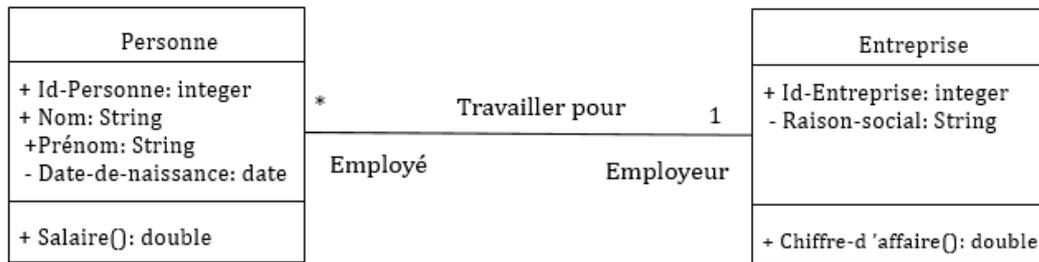


- Rôle d'association

Le rôle est une forme nominale, décrit la fonction d'une classe pour une association donnée. il est placé à une extrémité de l'association.



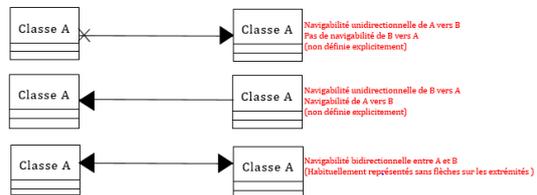
Exemple



- Navigabilité

La navigabilité indique si l'association fonctionne de manière unidirectionnelle ou bidirectionnelle, elle est représentée par une ou deux extrémités fléchées. La non-navigabilité se représente par un "X".

Les situations possibles de navigabilité sont représentées comme suit:



1.2.1. Formes d'association

a) Association binaire

Une association binaire est matérialisée par un lien entre deux classes associées.

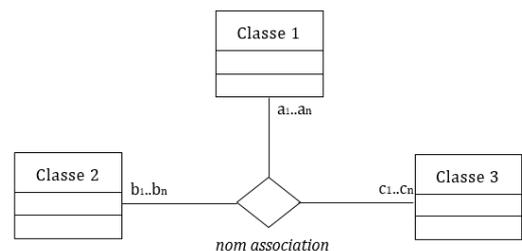


Représentation UML d'une association binaire

b) Association n-aire

Une association n-aire relie plus de deux classes. Elle est représentée en utilisant un losange reliant toutes les classes en question. Le nom de l'association, le cas échéant, apparaît à proximité du losange. Dans une association n-aire, la multiplicité apparaissant sur le lien de chaque classe s'applique sur une instance de chacune des classes à l'exception de la classe considérée.

Pour une association ternaire, les multiplicités se lisent de la façon suivante : Pour un couple d'instances de la classe 1 et de la classe 2, il y a C1 à Cn instances de la classe 3.

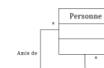


Représentation UML d'une association n-aire (ternaire)

c) Association réflexive

Une association est dite réflexive lorsque le lien existe entre des objets de la même classe.

Deux personnes peuvent être amies (on considère que l'amitié est réciproque).

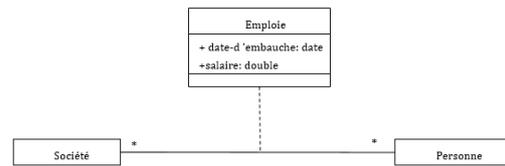


Représentation UML d'une association réflexive

d) Classe d'association

Une classe-association est une association qui est aussi une classe. Elle permet de décrire soit des attributs soit des opérations propres à l'association. Une classe-association est représentée par un trait en pointillé entre la classe et l'association qu'elle représente.

l'association "emploi" entre une société et personne possède comme propriétés le salaire et la date d'embauche. En effet ces deux propriétés n'appartiennent ni à la société, qui peut employer plusieurs personnes, ni aux personnes qui peuvent avoir plusieurs emplois.



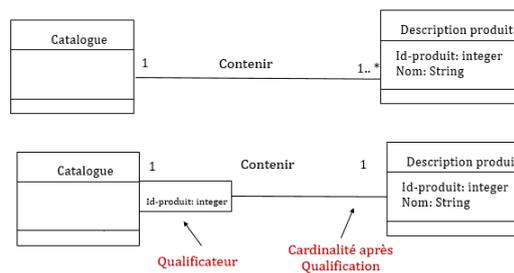
Représentation UML d'une classe-association

e) Association qualifiée

Une association qualifiée permet de restreindre la cardinalité d'une association en ajoutant un qualificateur (aussi appelé clé ou index). Ce qualificateur est constitué de un ou plusieurs attributs qui permettent de cibler un ou plusieurs objets en particulier. Le qualificateur est placé dans un rectangle à l'extrémité de l'association (extrémité opposée à la classe dont nous limitons la cardinalité).

Un objet de la classe Catalogue est relié à un nombre indéterminé d'objet de la classe Description produit.

Par contre, la classe Catalogue associée au qualificateur Id-produit n'est reliée qu'à un seul objet de la classe Description produit



Représentation UML d'une association qualifiée

f) Relation de dépendance

La dépendance entre deux classes permet de représenter l'existence d'un lien sémantique. Une classe B est en dépendance de la classe A si des éléments de la classe A sont nécessaires pour construire la classe B.

la relation de dépendance se représente par une flèche en pointillé entre deux classes.



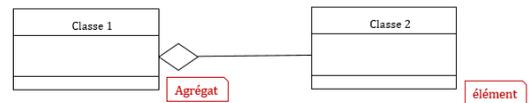
Représentation UML de la dépendance

1.3. Agrégation et composition entre classes

1.3.1. Agrégation

Une agrégation est une association qui permet de représenter un lien de type "ensemble" comprenant des "éléments". Il s'agit d'une relation d'inclusion structurelle ou comportementale d'un élément dans un ensemble.

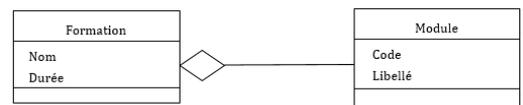
Une agrégation est noté graphiquement par un "losange vide" du côté de l'agrégat.



Représentation UML d'une agrégation

Exemple

Dans le cadre d'une formation, le cursus de cette dernière est une agrégation de modules à enseigner.

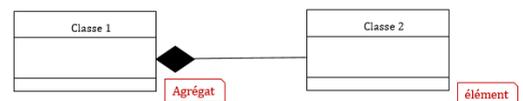


à noter que la suppression d'une formation ne conduit pas automatiquement à la suppression des modules étant donné que ces derniers peuvent très bien être enseignés dans d'autres formations.

1.3.2. Composition

Une composition également appelée agrégation composite ou agrégation forte, décrit une relation d'agrégation dans laquelle il existe une contrainte de durée de vie entre la classe "composant" et la ou les classes "composé". C'est-à-dire la destruction de la classe "composant" implique la destruction de la ou les classes "composé".

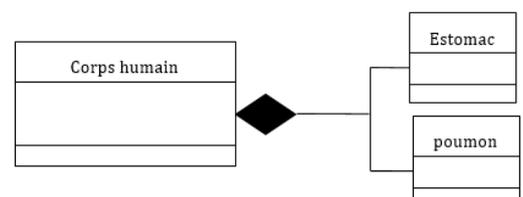
Une composition est noté graphiquement par un "losange plein" du côté de l'agrégat.



Représentation UML d'une composition

Exemple

Un corps humain est composé d'un estomac, des poumons, etc. le corps humain ne peut exister sans ses organes et réciproquement les organes ne peuvent exister en-dehors du corps humain.



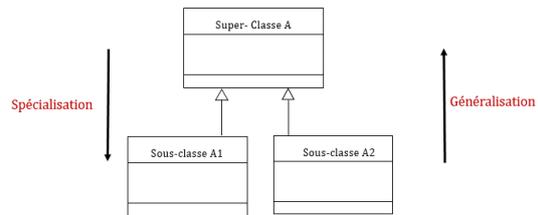
1.4. Généralisation et spécialisation

La généralisation décrit la factorisation des éléments communs (attributs et/ou opérations) d'un ensemble de classes dans une classe plus générale appelée super-classe (ou classe mère). Quant à la spécialisation représente le phénomène inverse, pouvoir dériver à partir d'une classe ou superclasse des sous-classes (ou classes filles) ayant des propriétés spécifiques les distinguant les une des autres.

Dans le langage UML, ainsi que dans la plupart des langages objets, ces deux propriétés se traduisent par le concept d'**héritage**.

Une relation d'héritage est noté graphiquement par une flèche avec trait plein dont la pointe est un triangle fermé désignant le cas le plus général.

les sous-classes A1 et A2 héritent de la super-classe A.



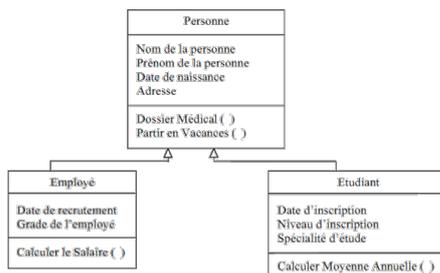
Représentation UML de l'héritage

Les propriétés principales de l'héritage sont :

- Une sous-classe possède toutes les caractéristiques de la super-classe, mais elle ne peut pas accéder aux caractéristiques privées de cette dernière ;
- Une sous-classe peut redéfinir une ou plusieurs opérations de la super-classe ;
- Toutes les associations de la classe mère s'appliquent aux classes dérivées ;
- Une instance d'une classe peut être utilisée partout où une instance de sa classe mère est attendue ;
- Une classe peut avoir plusieurs parents, c'est l'héritage multiple ;
- La relation d'héritage n'est pas propre aux classes. Elle s'applique à d'autres éléments du langage comme les acteurs, les cas d'utilisation ;
- La relation d'héritage est une relation non-réflexive, non-symétrique.

1.4.1. Héritage simple

L'héritage est simple lorsque une classe hérite d'une seule super-classe.



Les classes « Employé » et « Étudiant » sont des dérivés de la classe « Personne » et héritent des propriétés de cette dernière.

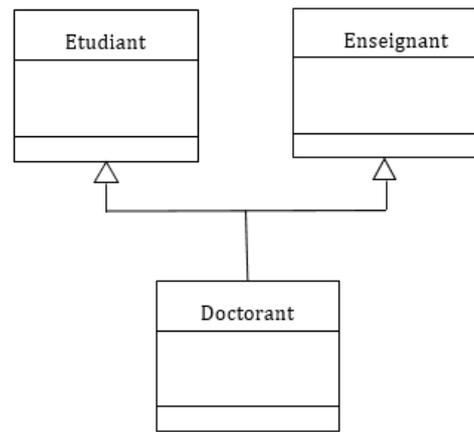
La classe « Employé » se distingue par d'autres attributs tels que la date de recrutement, le grade et le salaire

La « Étudiant » possède une date d'inscription, un niveau d'étude, une spécialité et une moyenne annuelle.

1.4.2. Héritage multiple

L'héritage est multiple quand une même hérite de deux ou plusieurs super-classes.

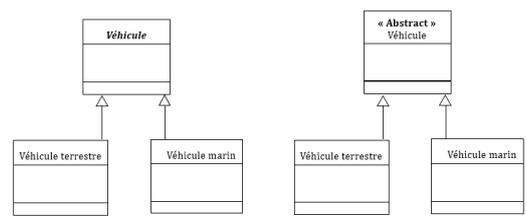
Un doctorant est à la fois un étudiant et un enseignant. Le doctorant hérite à la fois des propriétés des enseignants et de celles des étudiants



1.4.3. Abstraction

Parfois une classe ne peut exister que par l'existence de ses classes descendantes. Une telle classe est dite classe abstraite qui n'a pas d'instance directe, mais dont les classes descendantes ont des instances. Dans une relation d'héritage, la super-classe est par définition une classe abstraite.

Une classe abstraite peut être représenté soit en mettant son nom en italique, soit par rajout du stéréotype « abstract ».



Représentation UML d'une classe abstraite

1.5. Exercice 1

Il s'agit d'établir le schéma conceptuel d'une base de données pour la gestion des formations d'un institut privé :

- Un cours est caractérisé par un numéro de cours (NOCOURS), un libellé (LIBELLE), une durée en heures (DUREE) et un type (TYPE).
- Un cours peut faire l'objet dans l'année de plusieurs sessions identiques.
- Une session est caractérisée par un numéro (NOSES), une date de début (DATE) et un prix (PRIX).
- Une session est le plus souvent assurée par plusieurs animateurs et est placée sous la responsabilité d'un animateur principal.
- Un animateur peut intervenir dans plusieurs sessions au cours de l'année.
- On désire mémoriser le nombre d'heures (NBH) effectué par un animateur pour chaque session.
- Un animateur est caractérisé par un numéro (NOANI), un nom (NOMA) et une adresse (ADRA).
- Chaque session est suivie par un certain nombre de participants.
- Un participant est une personne indépendante ou un employé d'une entreprise cliente.
- Un participant est caractérisé par un numéro (NOPAR), un nom (NOMP) et une adresse (ADRP).
- Dans le cas d'un employé, on enregistre le nom (NOMEN) et l'adresse de l'entreprise (ADREN).

- On désire pouvoir gérer d'une manière séparée (pour la facturation notamment) les personnes indépendantes d'une part, et les employés d'autre part. Si nécessaire, on fera les hypothèses sémantiques complémentaires qui pourraient s'imposer.

Travail demandé: Établir un schéma conceptuel (diagramme de classes UML) pour cette base de données.

1.6. Exercice 2

Les stagiaires d'un établissement de formation désirent développer une application Web genre réseau social appelé netAtlas. Un membre du réseau netAtlas est identifié par une adresse e-mail et nom. Un membre peut avoir plusieurs amis qui sont aussi membres de netAtlas. Un membre peut effectuer une publication de contenu à une date donnée. Une publication concerne une ressource à publier, une ressource possède un nom et peut-être soit un message, soit une photo/vidéo soit un lien vers une page web. Une photo/vidéo a en plus du nom, une taille. Un lien vers une page web a une adresse (URL).

Travail demandé: Élaborez le diagramme de classe.