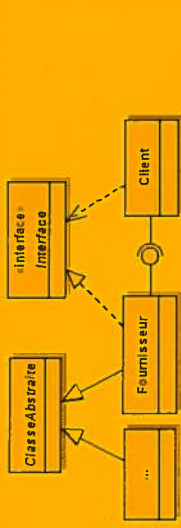


Diagramme de classes



Classe abstraite : classe qui ne s'instancie pas directement par opposition à une classe concrète. Elle sert en général à factoriser dans une super-classe des propriétés communes à un certain nombre de sous-classes et se note en italique.

Classe d'association : association propre au rang de classe. Elle possède tout à la fois les caractéristiques d'une association et d'une classe et peut donc porter des attributs qui se valorisent pour chaque lien entre objets.

Qualifieur (ou qualificatif) : attribut qui permet de « partitionner » l'ensemble des objets en relation avec un objet donné dans le cadre d'une association multiple.

Contrainte : condition portant sur un ou plusieurs éléments du modèle et qui doit être vérifiée par les éléments concernés. Elle est notée entre accolades {}, et souvent insérée dans une note graphique le point-ci.

Dépendance : relation sémantique entre deux éléments, dans laquelle la modification d'un des éléments (l'élément indépendant) peut affecter la sémantique de l'autre élément (l'élément dépendant).

Template : Classe paramétrée. Un rectangle en pointillés contenant une liste de paramètres formels est dessiné dans l'angle supérieur droit. Chaque paramètre possède un nom et un type ainsi qu'une valeur par défaut optionnelle.

Trucs et astuces

- En utilisant des interfaces plutôt que des classes, vous facilitez l'évolution de vos applications. Vous pourrez en effet changer l'implémentation des interfaces plus tard, sans aucun impact pour les classes qui les utilisent.
- Une classe possédant au moins une opération abstraite sans implémentation est forcément abstraite. Attention : une super-classe n'est pas toujours abstraite.
- Les classes d'association se rencontrent surtout dans le cas de multiplicités « * ». Par contre, il n'y aura qu'une seule instance de la classe d'association par couple d'objets reliés !
- Attention : l'utilisation d'un qualificatif d'association modifie la multiplicité à l'autre extrémité ! Celle-ci passe fréquemment de 0..* à 0..1.
- UML propose OCL (Object Constraint Language, mais autorise tout formalisme pour décrire les contraintes : texte libre, pseudo-code, Java, etc.
- Un lien durable entre objets va donner lieu à une association navigable entre les classes correspondantes ; un lien temporaire va donner lieu à une relation de dépendance. C'est le cas par exemple si une classe possède une opération dont un paramètre fait référence à une autre classe non reliée par une association.

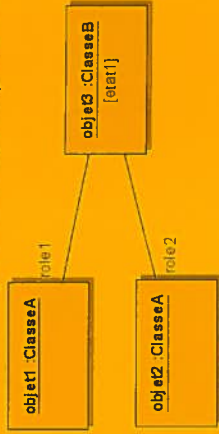
Diagramme de classes

- De nombreuses relations structurelles impliquent une dépendance ! C'est le cas pour l'association navigable : le client dépend du fournisseur. C'est également vrai pour la généralisation : la sous-classe dépend de sa super-classe.
- La règle générale est simple : minimisez les dépendances !
- N'utilisez pas forcément toutes les subtilités du diagramme de classes : pensez à votre lecteur !

Diagramme d'objets

Montre les instances des éléments structurels et leurs liens au run-time

Objet : entité aux frontières bien définies possédant une identité et encapsulant un état et un comportement ; un objet est une instance ou occurrence d'une classe. Un objet peut avoir connaissance d'un ou plusieurs autres objets : on parle alors de liens entre objets.



Trucs et astuces

- Le diagramme d'objets est un instantané, une photo, d'un sous-ensemble des objets d'un système à certain moment du temps, avec leurs propriétés et leurs relations. Il peut servir à donner un exemple pour un diagramme de classes complexe, à décrire une configuration initiale, etc.

Diagramme de packages

Montre l'organisation logique du modèle et les relations entre packages

Package : mécanisme général de regroupement d'éléments tels que classes, interfaces, mais aussi acteurs, cas d'utilisation, etc. Les packages peuvent être imbriqués dans d'autres packages.

Importation : relation de dépendance entre packages qui rend visibles les éléments publics d'un package au sein d'un autre package.

Trucs et astuces

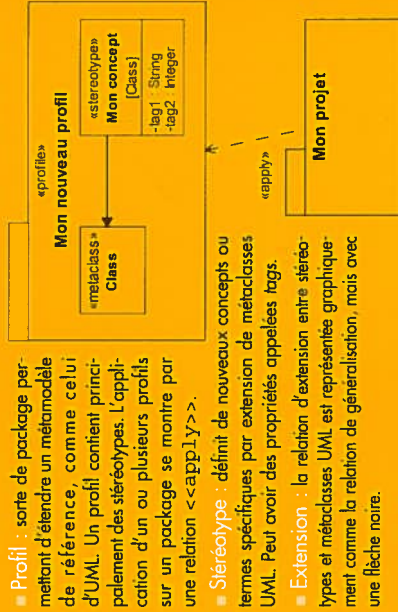
- Les packages sont des espaces de noms (ou namespaces) : deux éléments ne peuvent pas porter le même nom au sein du même package. Par contre, deux éléments appartenant à des packages différents peuvent porter le même nom. Du coup, UML propose une notation complète pour un élément : nomPackage : nomElement.
- Un modèle est représenté par un package avec le mot-clé « model » au-dessus du nom, ou avec un petit triangle en haut dans le coin droit du rectangle.

Diagramme de packages

- La structuration d'un modèle en packages doit s'appuyer sur deux principes fondamentaux : forte cohérence à l'intérieur de chaque package et faible couplage entre eux.
- Évitez au maximum les dépendances mutuelles entre packages ainsi que les dépendances circulaires !

Diagramme de profil

Définit un profil spécifique à un domaine ou une problématique par extension du métamodèle UML



- **Stéréotype** : définit de nouveaux concepts ou termes spécifiques par extension de métaclasses UML. Peut avoir des propriétés appelées tags.
- **Extension** : la relation d'extension entre stéréotypes et métaclasses UML est représentée graphiquement comme la relation de généralisation, mais avec une flèche noire.

Trucs et astuces

- Ce quatrième type de diagramme, introduit officiellement par UML 2.3, n'est pas utilisé par tout un chacun, mais uniquement par ceux qui souhaitent définir une variante d'UML pour un domaine spécifique.
- SysML (Systems Modeling Language) est un exemple récent de profil UML 2.

Diagramme de séquence

Montre la séquence verticale des messages passés entre éléments au sein d'une interaction



- **Ligne de vie** : représentation de l'existence d'un élément participant dans un diagramme de séquence.
- **Message** : élément de communication unidirectionnel entre lignes de vie qui déclenche une activité dans l'objet destinataire. La réception d'un message provoque un événement dans l'objet récepteur. La flèche pointillée représente un retour ou sens UML. Cela signifie que le message en question est le résultat direct du message précédent.

Trucs et astuces

- La notation de base du diagramme de séquence est très simple : elle suffit déjà dans de nombreux cas et présente l'avantage de ne pas nécessiter d'explication !