

Modélisation UML

Historique

- Pour faire face à la complexité croissante des systèmes d'information, de nouvelles méthodes et outils ont été créés.
- La principale avancée réside dans la programmation orientée objet (P.O.O).
- Face à ce nouveau mode de programmation, les méthodes de modélisation classiques (telle que MERISE) ont rapidement montré certaines limites et ont dû s'adapter.
- Dans ce contexte, et devant le foisonnement de nouvelles méthodes de conception orientée objet, l'Object Management Group (OMG) a eu comme objectif de définir une notation standard utilisable dans les développements informatiques basés sur l'objet.
- C'est ainsi qu'est apparu UML qui est issu de la fusion des méthodes Booch, OMT (Object Modelling Technique) et OOSE (Object Oriented Software Engineering).

L'approche objet

Le concept objet, c'est :

- Une entité aux frontières précises qui possède une identité (nom).
- Un ensemble d'attributs qui caractérise l'état de l'objet.
- Un ensemble d'opération (méthodes) qui définit son comportement.
- Un objet est une instance de classe (occurrence d'un type abstrait).
- Une classe est un type de données abstrait caractérisé par des propriétés communes à des objets et permettant de créer des objets possédant ces propriétés.

Définir UML

UML : Unified Modeling Language → langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système.

Modèle : abstraction de la réalité. C'est une vue subjective mais pertinente de la réalité.

UML c'est :

- Une norme.
- Un langage de modélisation objet.
- Un support de communication.
- Un cadre méthodologique pour une analyse objet.

→ UML est une boîte à outils qui permet d'améliorer progressivement ses méthodes de travail tout en préservant les modes de fonctionnement.

→ Chacun utilise les outils qui lui sont utiles

Il s'agit d'un langage formel et normalisé qui allie précision, gage et stabilité. Sa mise en pratique nécessite un apprentissage et passe par une période d'adaptation.

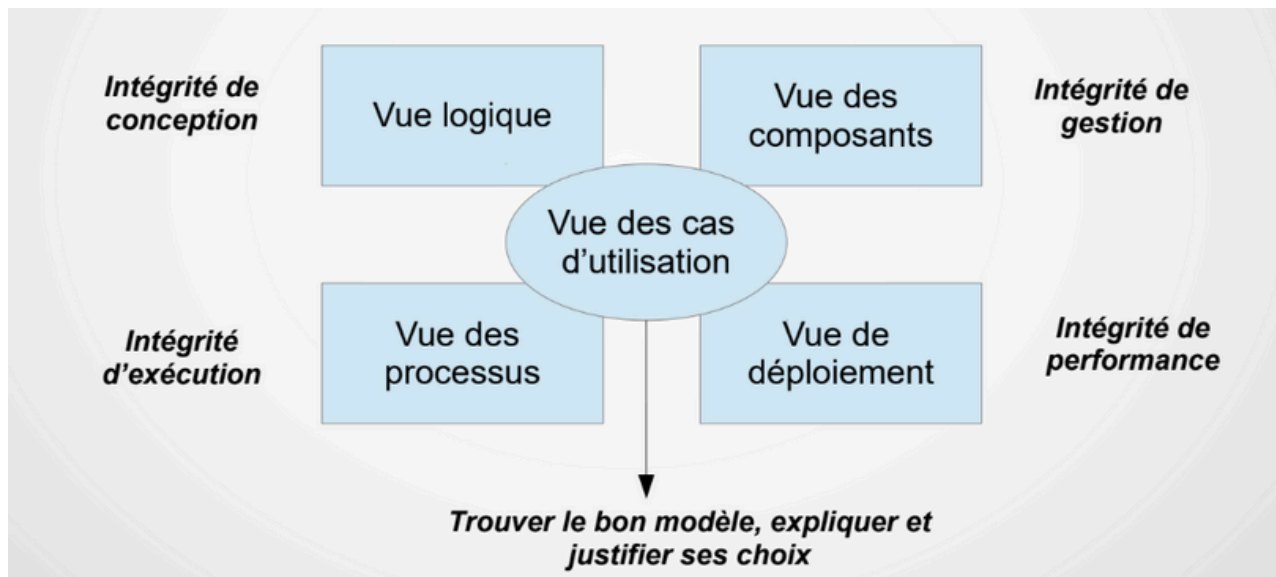
Démarche à suivre

Les auteurs de UML préconisent d'utiliser une démarche :

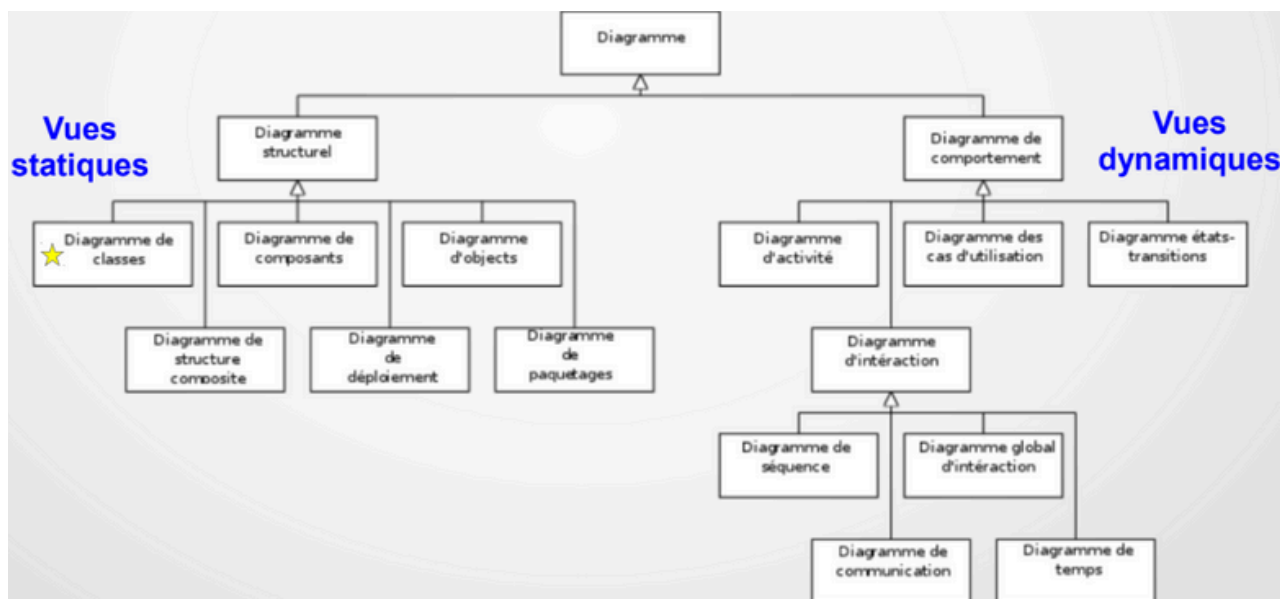
- Itérative et incrémentale : mieux vaut s'y prendre en plusieurs fois, en affinant son analyse par étapes.
 - Guidée par les besoins des utilisateurs : les utilisateurs définissent ce que doit être le système.
 - Centrée sur l'architecture logicielle : le schéma 4+1 vues de KRUCHTEN.
-

Modélisation UML

La vue 4+1 de KRUCHTEN



Les 14 diagrammes UML existants



Le diagramme de classes

Il exprime la structure statique entre termes de classes et de relations entre celles-ci.

Il permet de modéliser les entités des systèmes d'informations à l'aide de 6 concepts :

- Classe
- Attribut
- Identifiant
- Relation
- Opération
- Généralisation / Spécialisation

Modélisation UML

Les concepts

- Classe : représentation abstraite d'un ensemble d'objets : elle définit leur structure, leur comportement et leurs relations.
- Attribut : information élémentaire représentée par son nom et son format.
- Identifiant : attribut particulier qui permet de repérer de façon unique un objet (souligné).
- Opération : fonctionnalité assurée par la classe.

NOM CLASSE
<u>Attribut_1</u> : int Attribut_2 : string Attribut_3 : int
Operation_1() : void Operation_2() : void

La notion de relation

S'il existe des liens entre les objets, cela se traduit obligatoirement par des relations qui existent entre leurs classes respectives.

Il existe plusieurs types de relations entre classes :

- L'association.
- La généralisation / spécialisation.
- La dépendance.

L'association

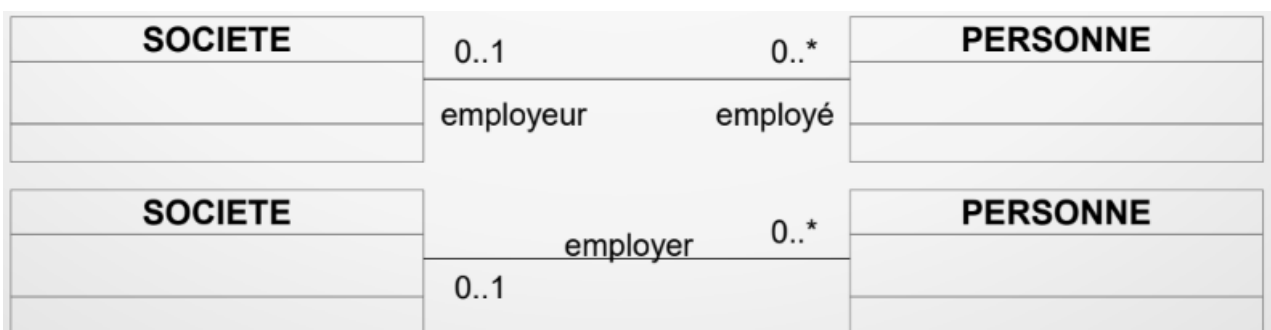
C'est la relation la plus courante.

Elle fait apparaître la multiplicité (ou cardinalité) :

- 1..1 : un et un seul.
- 0..1 : zéro ou un.
- N ou * : N (entier naturel).
- M..N : de M à N (entiers naturels).
- 0..* : de zéros à plusieurs.
- 1..* de 1 à plusieurs.

La multiplicité est notée à l'envers de la notation MERISE.

On peut mettre un nom ou un verbe pour faciliter sa lecture (optionnel).

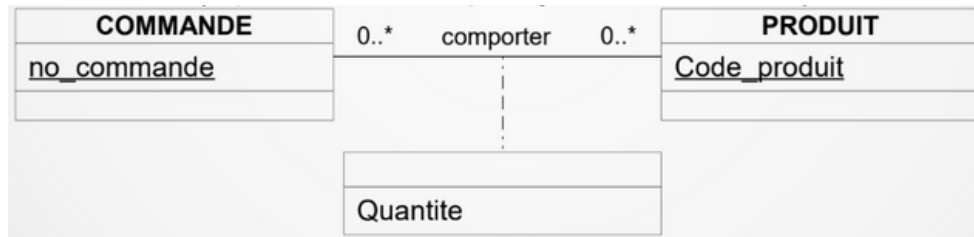


Modélisation UML

Les classes-associations

Parfois un attribut dépend fonctionnellement de 2 identifiants appartenant à 2 classes différentes.

L'association est alors dite porteuse d'attribut et est appelée classe-association (il peut aussi ne pas y avoir d'attribut).



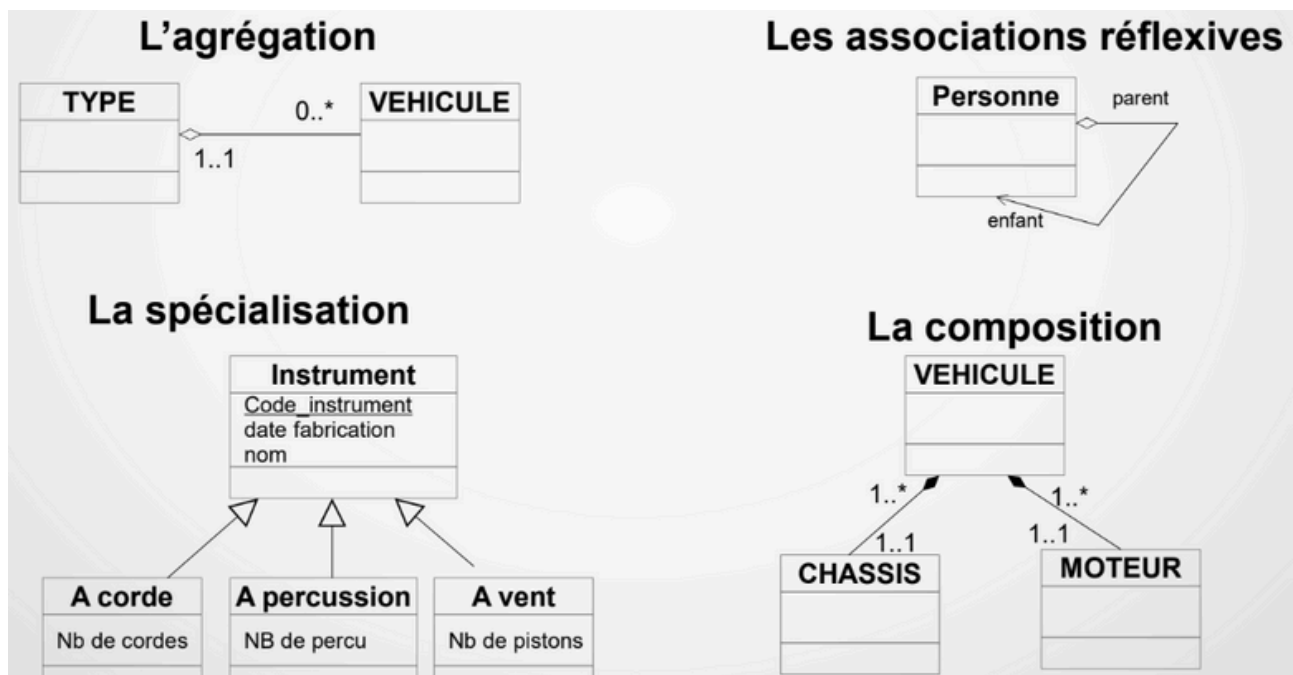
Toute classe-association peut être remplacée par une classe intermédiaire qui sert de pivot pour une paire d'association.



Autres variantes

Parfois un attribut dépend fonctionnellement de 2 identifiants appartenant à 2 classes différentes.

L'association est alors dite porteuse d'attribut et est appelée classe-association (il peut aussi ne pas y avoir d'attribut).



Modélisation UML

Cas des données SIG

Il n'existe pas à proprement parler de règles pour les données SIG.

Il y a 4 scénarios possibles :

- Faire apparaître les données SIG comme spécialisation de Geometry (choix fait dans certains modèles COVADIS).

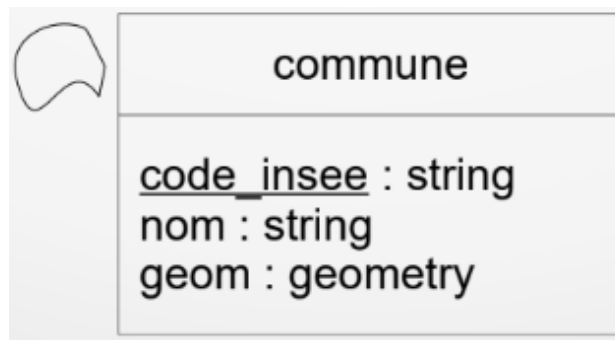
→ Peu lisible.

- Dessiner un pictogramme pour préciser la nature de la géométrie.

→ Ne gère pas les classes qui auraient plusieurs géométries + ne permet pas de connaître le nom de l'attribut géométrique.

- Gérer l'information géométrique comme n'importe quel attribut.

- Mixte des deux dernières solutions.



Construire un diagramme de classes



Les questions à se poser :

- Quelles sont toutes les informations que l'on souhaite stocker ?
- Quels regroupements d'informations à stocker en classes naturelles peut-on faire ?
- Quelles informations peuvent servir d'identifiants à valeur unique ?
- Est-ce qu'il y a des doublons d'informations ?
- Est-ce qu'il y a des calculs à réaliser ?
- Est-ce qu'une information peut être trouvée parmi une liste de choix ?
- Quels sont les points d'attention sur les classes-associations ?
- Quels sont les automatismes de réalisation ?
- Comment est gérée l'historisation des actions ?
- Est-ce qu'il s'agit d'un modèle évolutif ?

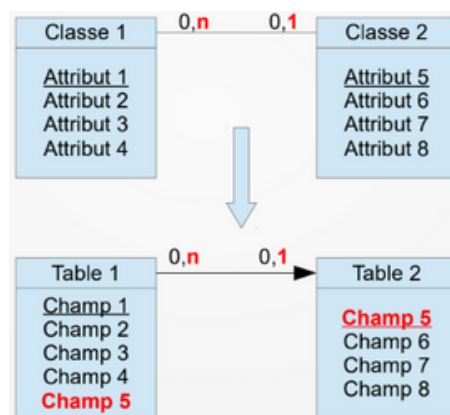
Modélisation UML

Le Modèle Physique de Données (MPD)

Modèle physique de données	Modèle physique de données
Classe	Table
Attribut	Champ
Occurence	Enregistrement
Clef primaire	Clef primaire
Association	<u>1 et n :</u> L'identifiant migre de 1 vers n et devient clef étrangère. <u>n et n :</u> Les deux identifiants migrent et deviennent à la fois clefs primaires et étrangères.

Passer d'un diagramme de classes à un MPD

L'identifiant migre de 1 vers n et devient clef primaire :



Les deux identifiants migrent et deviennent à la fois clefs primaires et étrangères :

