

# 62-41.2 Ingénierie pilotée par les modèles de données

## Généralités MDE

[MDE-01.02]

*Semestre Printemps 2022-2023*

Fabrice Camus

[fabrice.camus@he-arc.ch](mailto:fabrice.camus@he-arc.ch)

## Production de logiciels (de gestion)

Mesure du succès → par la satisfaction du client

- Court terme : objectifs, délais, coûts
- Long terme : réactivité par rapport à l'évolution de l'environnement → capacité à faire évoluer les logiciels de gestion (agilité)

### Qualité vs Excellence

*« La qualité n'est pas l'excellence d'un produit ou service, mais son aptitude à satisfaire les besoins de l'acheteur ou consommateur. Un produit de qualité n'offre ni plus ni moins que ce qui en est attendu ».* [P-A. Sunier]

### Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

# Ingénierie

## Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

« Étude d'un projet industriel sous tous ses aspects (techniques, économiques, financiers, monétaires et sociaux) et qui nécessite un travail de synthèse coordonnant les travaux de plusieurs équipes de spécialistes. »

[Larousse]

« L'ingénierie est une activité scientifique et rigoureuse de conceptualisation et réalisation des ouvrages d'art fonctionnels, dans la construction des ensembles structuraux, mécaniques, chimiques, [...] ou électroniques et qui s'exécutent dans les règles de l'art et de la rigueur scientifique. Les principes sur lesquels reposent l'ingénierie et sa méthodologie sont éminemment logiques. Ses référents sont d'ordres technique, scientifique et mathématique. »

[wikipedia]

« L'ingénierie est l'art de mener méthodiquement les actions pertinentes et nécessaires à transformer une idée souvent originale et/ou ambitieuse en un produit ou service adéquat aux plans économique, social et environnemental. »

[P.-A. Sunier]

# Modélisation

## Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

## Modèle ? Rappel....

- Représentation simplifiée d'une réalité
- Selon un point de vue, un niveau d'abstraction
- Pour un but donné
- Respectant un formalisme (sémantique + notation textuelle ou graphique)

## Exemples

- Modèle conceptuel de données
- Diagramme de classe UML
- Plans des canalisations d'une maison
- Plans du système électrique
- Comptabilité (modèle des flux financiers)
- ...
- On s'intéresse bien sûr aux langages de modélisation

# Modélisation

## Pourquoi modéliser ?

- Complexité des organismes et de leur SI → Faciliter la compréhension et la description des systèmes
- Immatérialité des logiciels de gestion → Percevoir, faire prendre conscience de la complexité informatique
- Anticiper les problèmes
- Outils de communication
- ...

# Modélisation

## Méta-modèle

- Un modèle doit être conforme à son méta-modèle
- Un méta-modèle décrit, spécifie formellement les concepts communs à un langage.
- Un méta-modèle décrit le langage de modélisation
- Les concepts du méta-modèle possèdent tous une sémantique précise

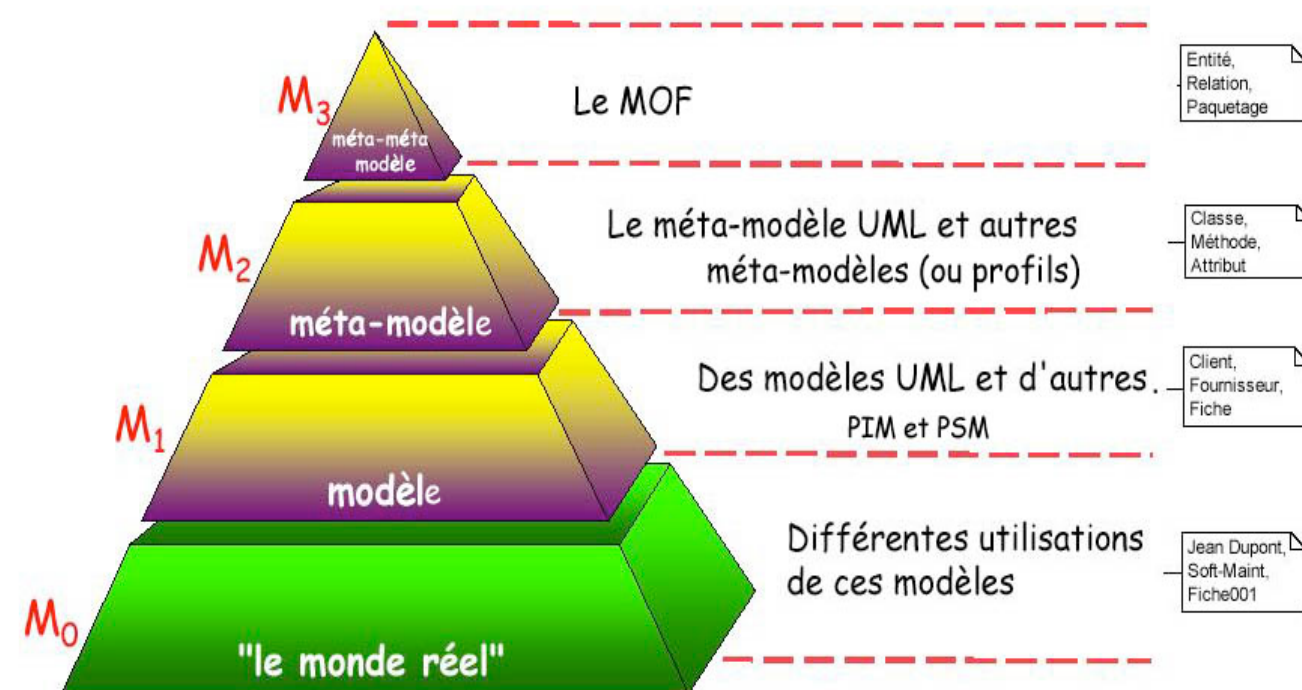
## Exemples

- Un diagramme de classe UML doit être conforme au méta-modèle UML
  - Concepts de classes, propriétés, opérations, associations...
- Un MCD doit être conforme à son méta-modèle, le modèle Entité-Associations
  - Concepts d'entité, d'association, d'attribut, d'identifiant...
- Un document XML doit être conforme à son schéma XSD



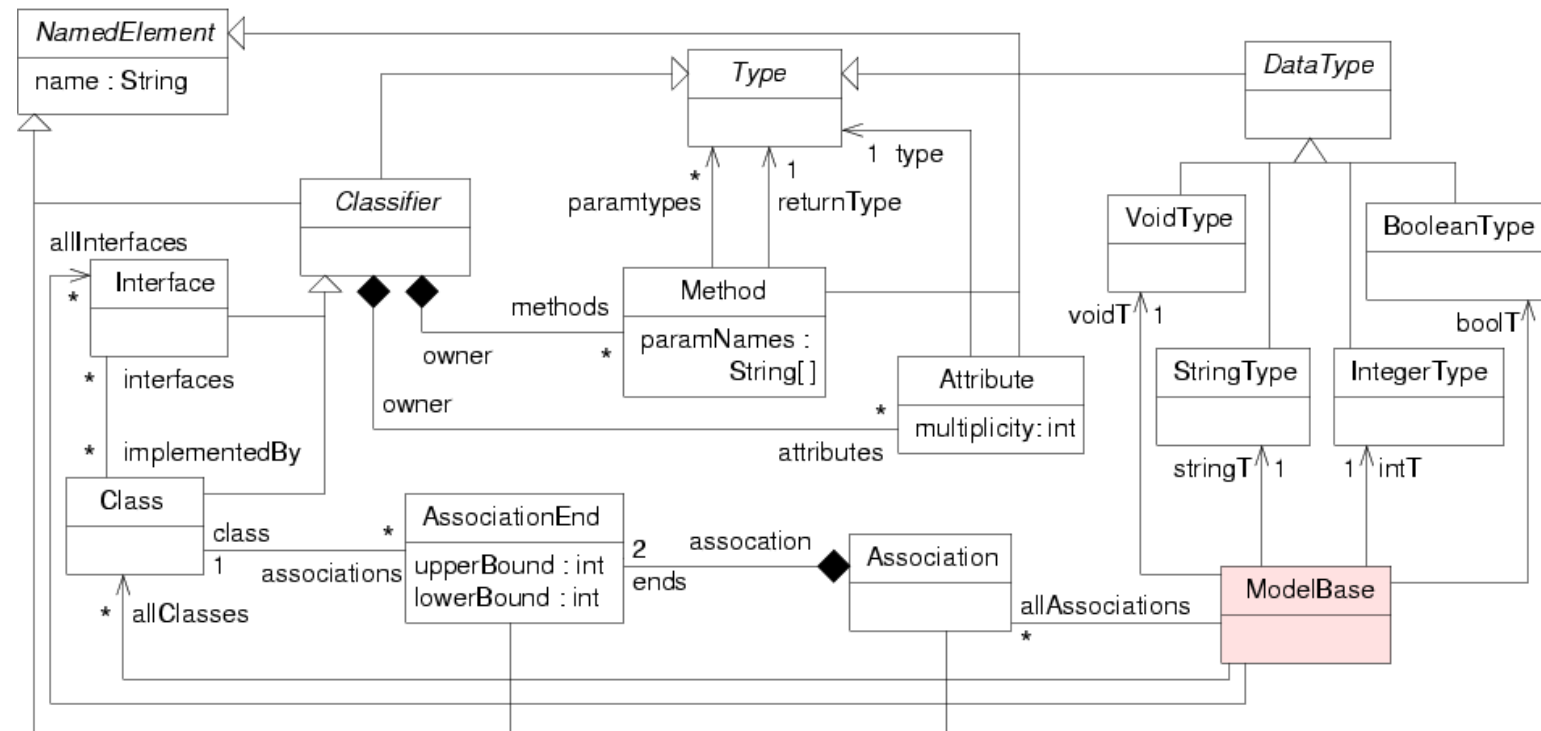
# Modélisation

## Architecture à 4 niveau



Source : Prof. J. Béziniv illustrant MDA et les standards de l'OMG

## Extrait du méta-modèle UML





## Emphase sur les données

### Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

Les données sont stables

- La nature (structure) des données d'une entreprise va rester stable dans le temps

Les traitements (processus) le sont moins

- Evolution des processus de manipulation de données (acquisition, traitement, persistance, restitution...)

Quelque soit la méthodologie retenue (classique ou objet), on privilégie de mettre l'effort sur la modélisation des données

Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

## Diviser pour régner

Diviser un problème de grande taille en plusieurs sous-problèmes de petite taille et plus abordable

Toute démarche d'analyse préconise le découpage en sous-système

Réduire la complexité (mathématique)

# Automatisation

## Concepts-clé (non-exhaustif)

- Ingénierie et modélisation
- Emphase sur les données
- « Diviser pour régner »
- Automatisation

## Automatiser les activités compliquées

- Activités mécanistes → Sans valeur ajoutée
- Activités mécanistes → Ensemble fini de règles connues

## Gages de qualité

- Répétitivité
- Fiabilité
- Rapidité

## Exemples

- Transformation MCD-MLD
- Génération de (squelettes de ) code

## MDE

### Model-Driven Engineering

*« L'ingénierie dirigée par les modèles (MDE) est la discipline informatique mettent à disposition des outils, concepts et langages pour créer et transformer des modèles.*

*Il s'agit d'une ingénierie générative, par laquelle tout ou partie d'une application est générée à partir de modèles servant de spécifications.*

*Cette discipline propose donc de mécaniser les processus que suivent les ingénieurs à la main. »*

[\[wikipedia\]](#)

→ Les modèles servent de base de spécifications

→ Déléguer la production des artefacts logiciels à des automates / générateurs de code

On parle donc de :

- Transformation automatisée de modèles en d'autres modèles
- Génération de code

## MDE

Les modèles servent de base de spécifications

Déléguer la production des artefacts logiciels à des automates / générateurs de code

On parle de :

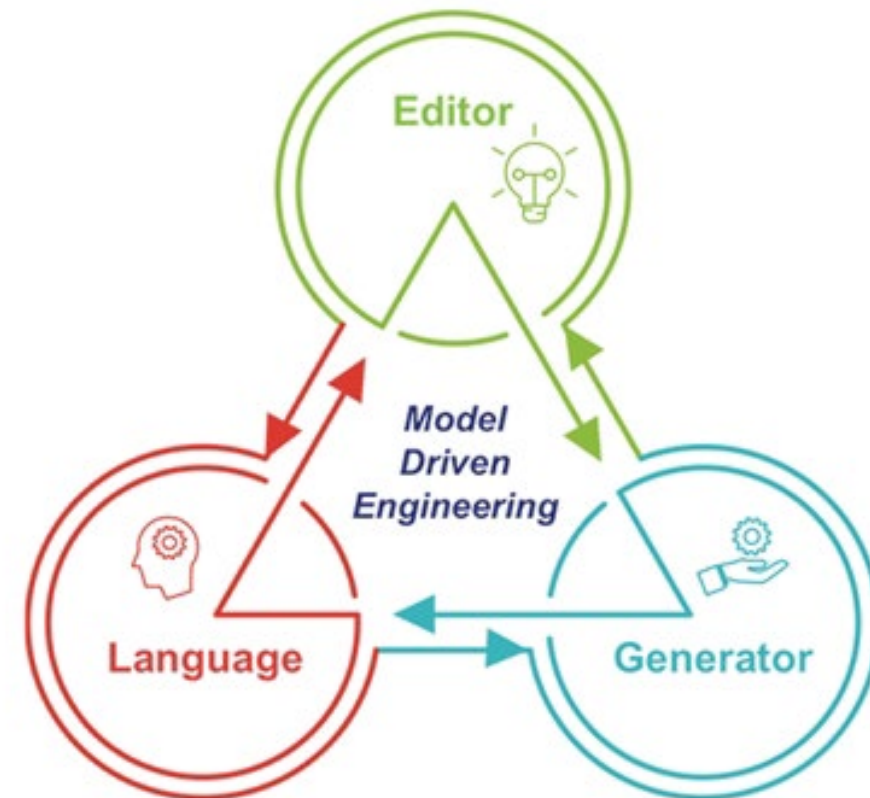
- Transformation automatisée de modèles en d'autres modèles
- Génération de code

## MDE

Principe 1 : Les modèles doivent être riches, précis et sans ambiguïté pour être transformables.

Principe 2 : Les modèles s'appuient sur un DSL, *Domain Specific Language*, méta-modèle riche, enrichi, pour permettre la transformation des spécifications. Le DSL peut exister sur le marché ou être créé spécifiquement par l'entreprise

Principe 3 : L'environnement propose un éditeur et des outils de transformation s'appuyant sur le DSL

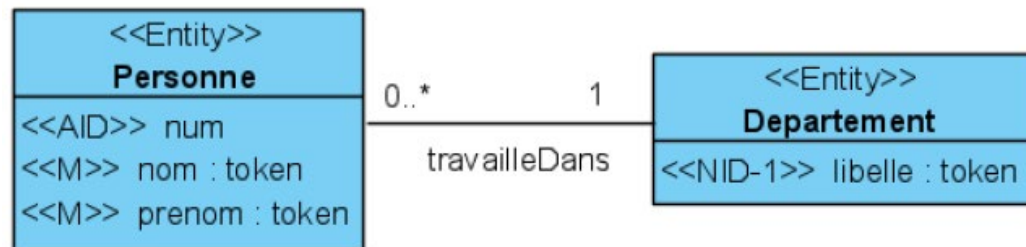


source : [The Anatomy of Model Driven Engineering | MDE Systems](#)

## Champs d'applications concrets

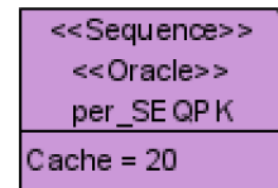
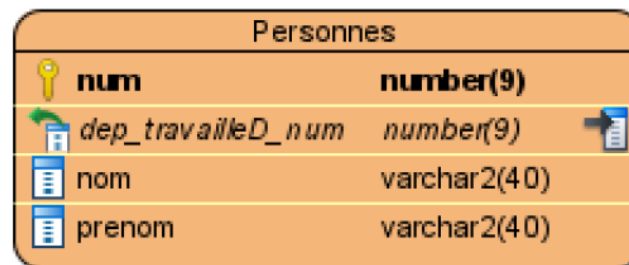
### Modélisation des données

- Du MCD au code SQL-DDL



```

CREATE SEQUENCE per_SEQPK;
CREATE TABLE Personnes (num number(9) NOT NULL, ... );
  
```

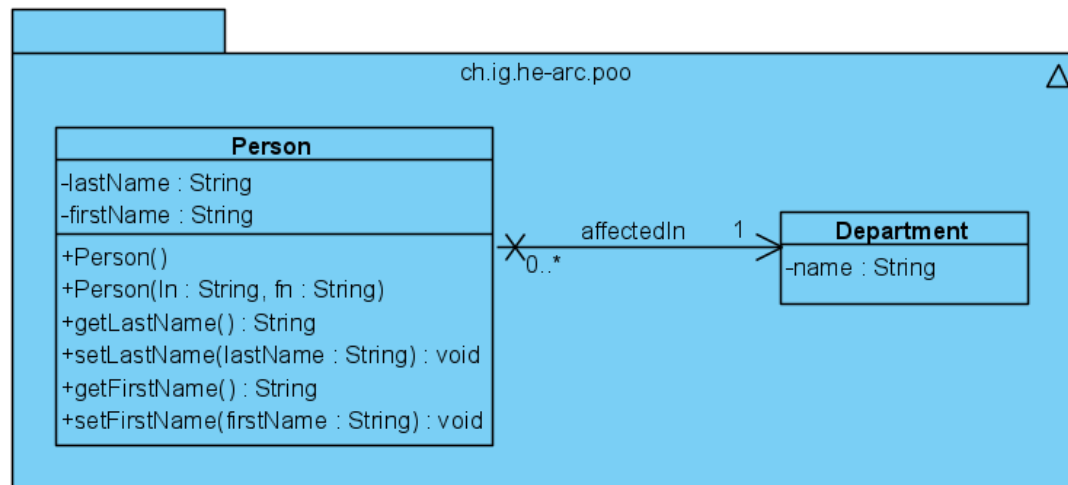




## Champs d'applications concrets

### Diagramme de classe «Objet»

- Génération de (squelette de) code (Java ou autre)



#### Code Java

```
package ch.ig.he-arc.poo;
public class Person {
    private String lastName;
    private String firstName;
    private Department dep

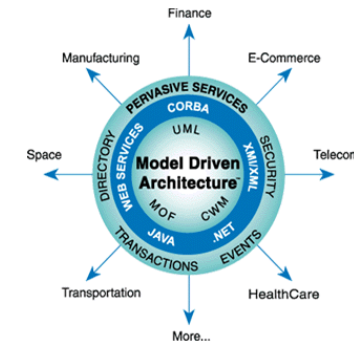
    public Person(String ln, String fn) {
        lastName=ln;
        firstName=fn;
    }

    public String getLastName() {
        return this.lastName;
    }
    ...
}
```

## MDE avec MDA

### MDA®, Model-Driven Architecture

- Adopté et publié par l'OMG® (ce n'est pas un standard)
- Approche de développement de logiciel
- L'idée est de résoudre les problèmes d'interopérabilité et de portabilité dès le niveau modélisation → Les modèles au cœur du processus



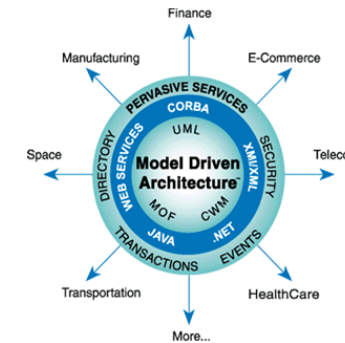
### Objectifs

- Pérenniser les modèles métiers
- Prendre en compte les plateformes d'exécution, pour garantir une meilleure interopérabilité, portabilité, réutilisabilité des composants
- Gains de productivité, par des processus de transformations et générations

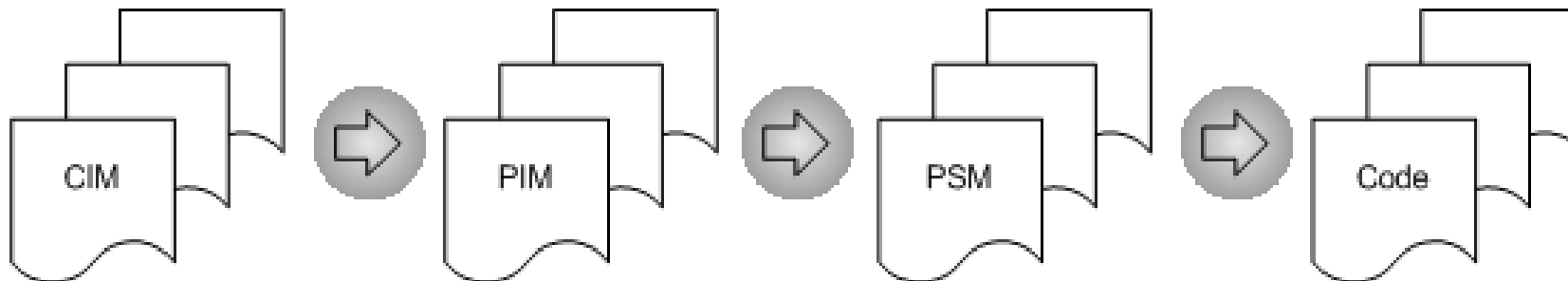
## MDA (2)

Les deux concepts-clés

- Séparer les aspects métiers des aspects techniques
- Proposer des transformations dans des plateformes cibles spécifiques



Mise en œuvre



CIM : Computation Independant Model

PIM : Platform Independant Model

PSM : Platform Specific Model

## MDA (3)

S'appuie sur les standards de l'OMG

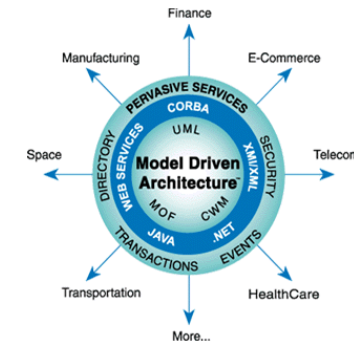
- MOF, UML & les profils, OCL, XMI, CWM

Quelques standards en cours d'élaboration ou "obscurs"

- AS, DI, fUML, ALF, QVT...

MDA est une belle intention, un concept

- Peu d'outils sur le marché
- Aucune préconisation quant à l'élaboration des modèles
- Standards : Complexité, interaction, évolution très lente
- UML et ses « lacunes »



## D'autres mises en œuvre

### Microsoft

- fourni des outils pour développer ses propres DSL et générateurs de code

Quelques sociétés commercialisent leur produits

### P. A. Sunier & Filière IG

- MVC-CD

## AGL vs Outils CASE

AGL : Atelier de Génie Logiciel

CASE : Computer-Aided Software Engineering

*« Un outil CASE est un outil d'ingénierie du logiciel qui se traduit par environnement informatisé, composé d'outils d'automatisation, respectant une méthode et s'appuyant sur un référentiel. Ces outils sont intégrés au CASE qui en assure leur coopération de façon uniforme et transparente. »*

*« Un AGL (dans une vision francophone) est un outil CASE, complété par des outils nécessaires au travail annexe à celui de la pure conception et du développement. Les outils bureautiques et de conduite de projet en sont des bons exemples. Il offre donc un environnement de travail complet et formalisé. »*

*[F. Camus]*

## AGL & Outils CASE

Outils MDE des années 1980-2000, propriétaires

Différentes catégories et couverture fonctionnelle

- Upper-Case, Lower-Case, intégré

Basés sur l'approche fonctionnelle/classique

N'ont pas pu/su prendre le virage des années 2000 (Objet, Open Source, Internet) et les effets de mode

... mais les concepts sont toujours en vigueur !!!



## A retenir

Domaine de l'ingénierie du logiciel

Ingénierie dirigée/pilotée par les modèles

- Ingénierie générative
- Pilotage par les modèles

Ensemble d'outils (intégrés ou non) et de méthodes

Outils d'aide

Couverture du cycle de vie

Automatisation de tâches

## Concepts-clés dans une démarche MDE

Référentiel

Modèle vs Diagramme

Postulat d'utilisation des générateurs

## Le Référentiel

**Pièce maitresse** dans un processus d'IDM

**Stocke** les différents objets de **définition** d'un SI

Les restitue sous forme graphique, textuelle ou autre

Manipulé par les **outils** de l'agl

Assure la **transition**, la **traçabilité** des objets entre les **niveaux d'abstraction**

## Modèle vs Diagramme

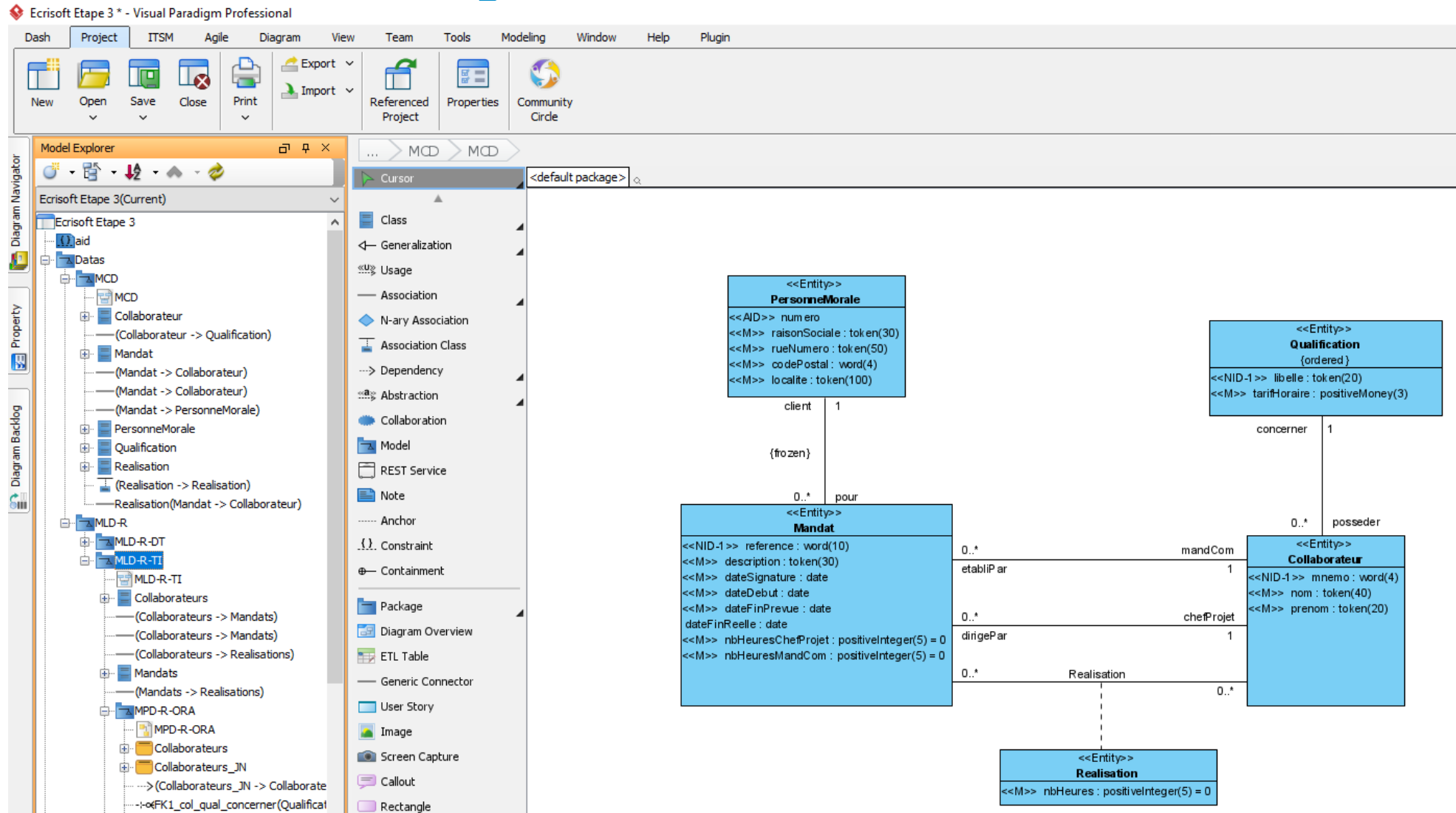
Modèle : Ensemble des spécifications

- Les objets de définition
  - Exemple : les entités CLIENT et COMMANDE
- Plusieurs modèles dans un projet !

Diagramme : vue externe du référentiel

- Un diagramme est aussi un objet du référentiel
- Mais le diagramme sert à restituer graphiquement un objet du référentiel
- Exemple : le diagramme intitulé MCD, affichant les entités CLIENT et COMMANDE
- Un même objet peut apparaître sur plusieurs diagrammes, il n'existe pourtant qu'une seule fois dans le référentiel
- Que se passe-t-il si je supprime un diagramme ?

## Exemple avec Visual Paradigm



## Postulat d'utilisation des générateurs

A LIRE sur cyberlearn :

- Postulat d'utilisation des générateurs