

Interopérabilité des systèmes d'information

4. *Diagrammes pour les modèles conceptuels de données*

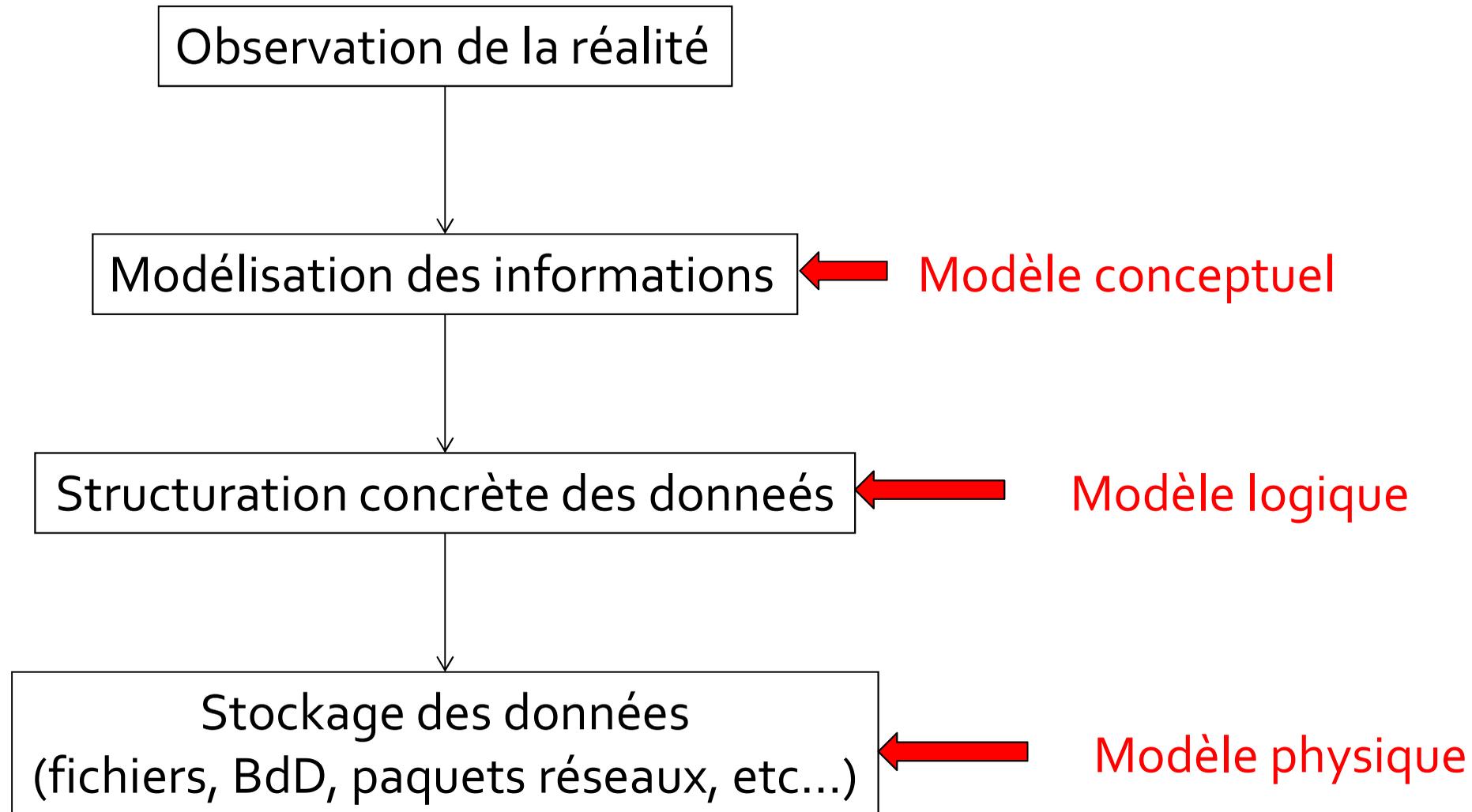
4.2 Diagrammes de classes UML

Mastère Management de la transition industrielle

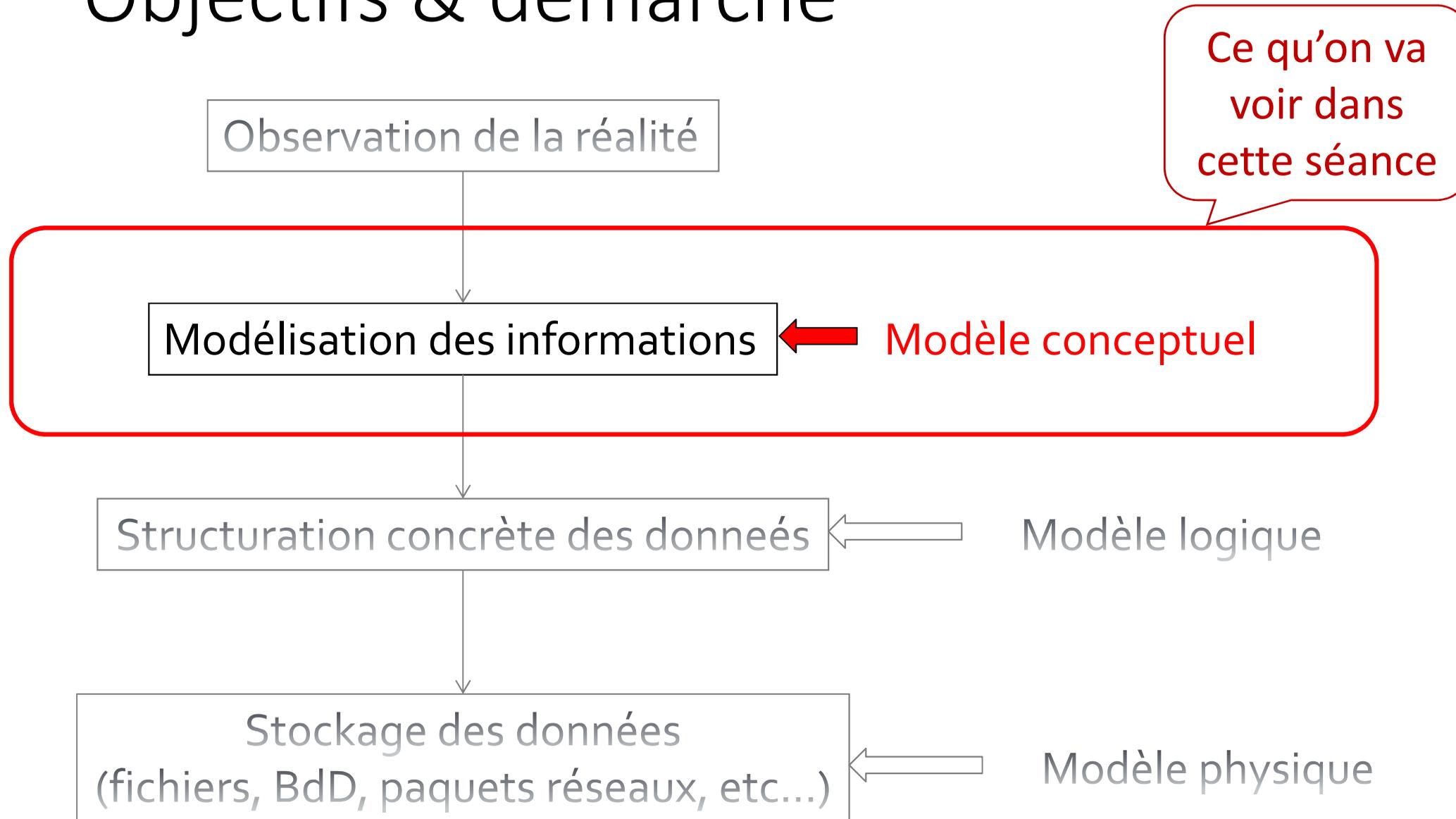
Antoine Zimmermann

9 février 2024

Objectifs & démarche



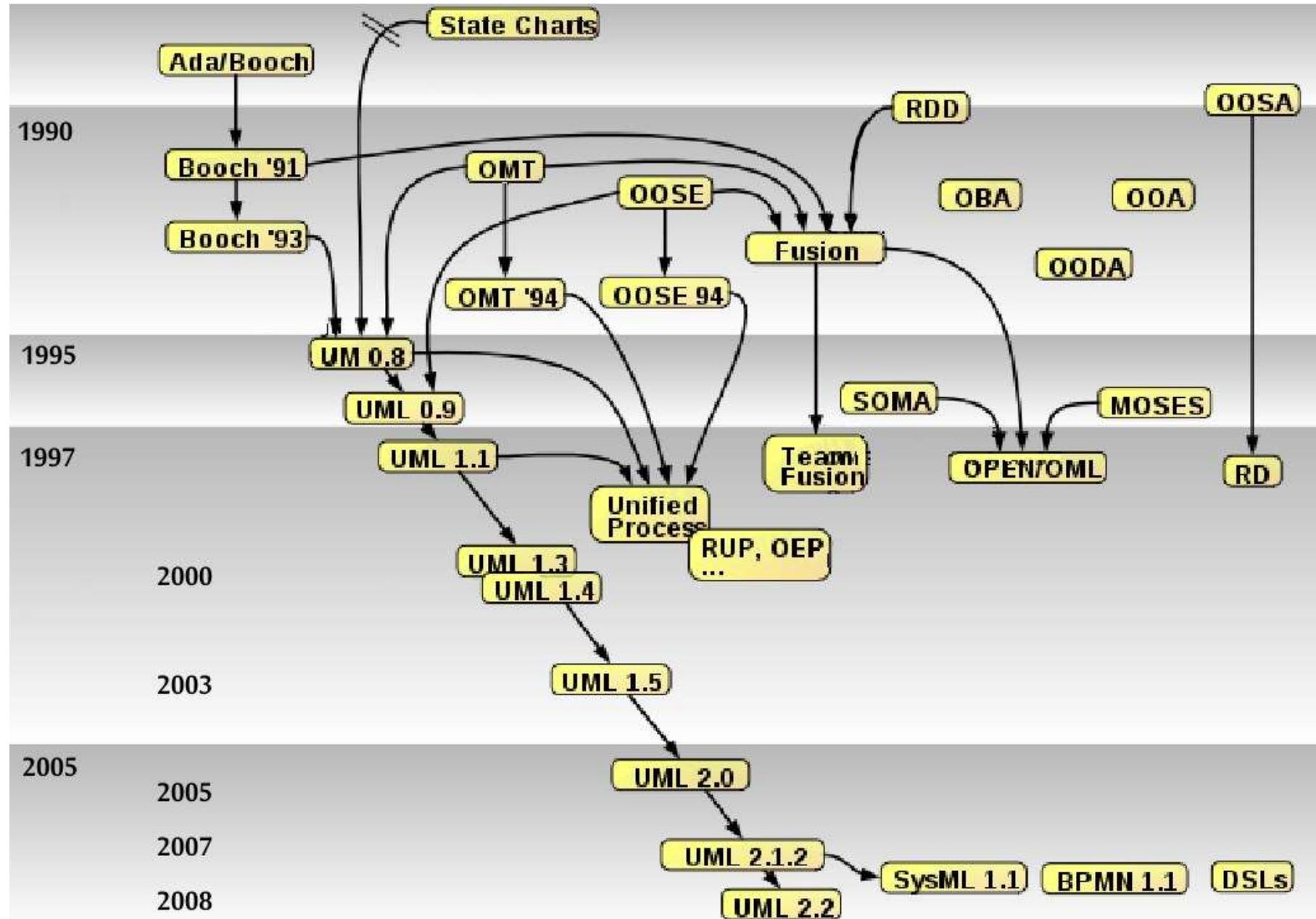
Objectifs & démarche



UML: *Unified Modelling Language*

- En français, le **langage de modélisation unifié**
- Langage **graphique** (sous forme de diagrammes) permettant de décrire la conception et le fonctionnement de logiciels ou d'architectures informatiques, indépendamment d'un langage de programmation ou d'une mise en œuvre concrète.
- Utilisé pour décrire:
 - Les activités d'un objet/logiciel
 - Les acteurs
 - Les processus
 - Les schémas de base de données
 - Les composants logiciels
 - La réutilisation de composants

Évolution d'UML et son écosystème



Les diagrammes de classes

- Initialement conçus pour décrire les structures de données et éléments logiciels de types *classes* et *interfaces* en **programmation orientée objet**
- Aujourd'hui, les diagrammes de classes sont utilisés aussi comme modèle conceptuel de données ou même comme formalisme de représentation des connaissances

Comparaison avec modèle EA

- Une classe UML = un type d'entité EA
- Attribut de classe \approx attribut d'entité
- Relation entre classe \approx association
- Cardinalité de relation = cardinalité d'association
- UML ajoute de nombreuses choses:
 - Visibilité des attributs
 - Méthodes (\approx fonctions) qu'une classe ou ses instances peut exécuter
 - Plusieurs types de relation (héritage, association, agrégation, composition)
 - Classes abstraites, interfaces, templates

Exemple 1

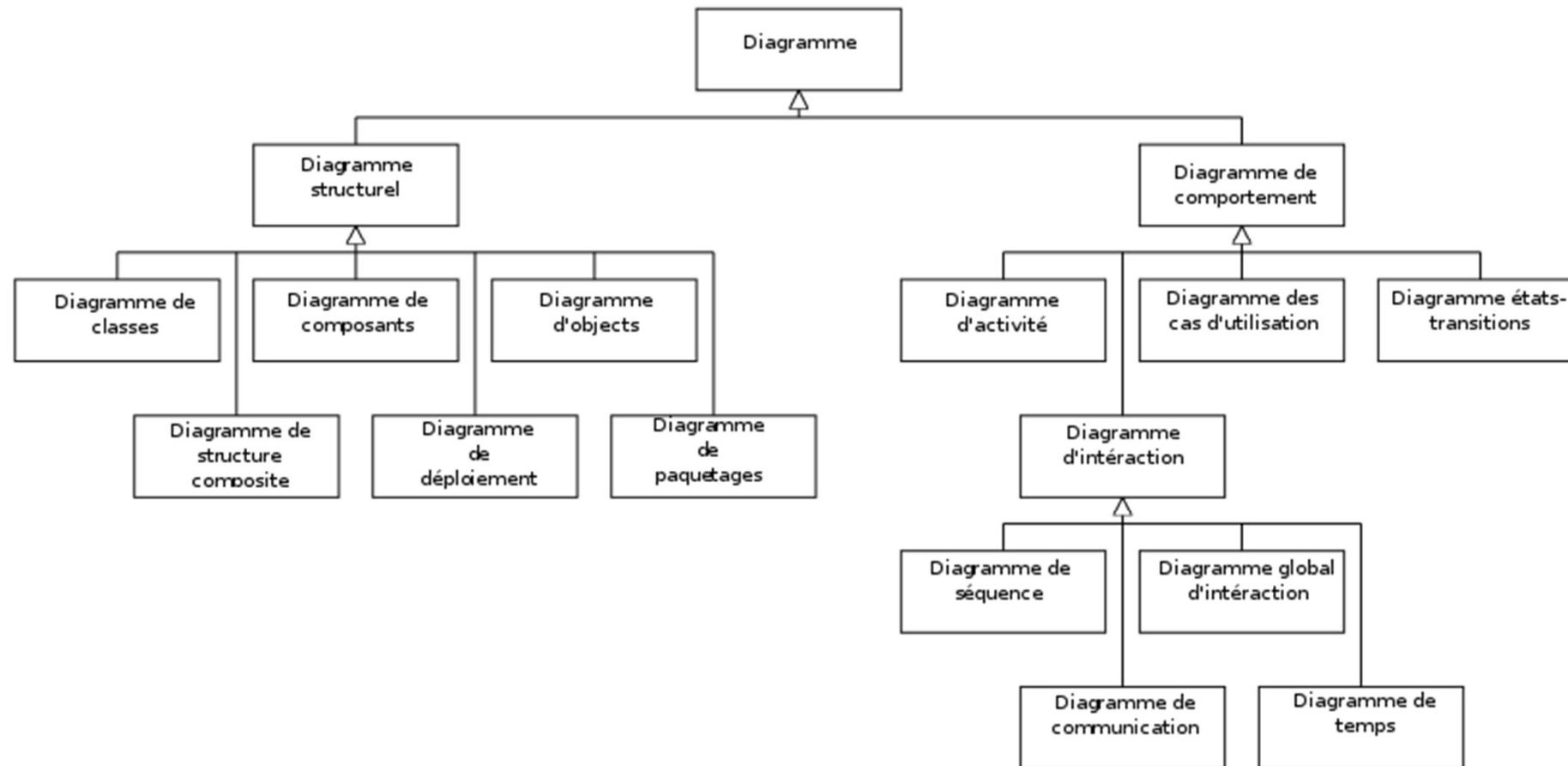
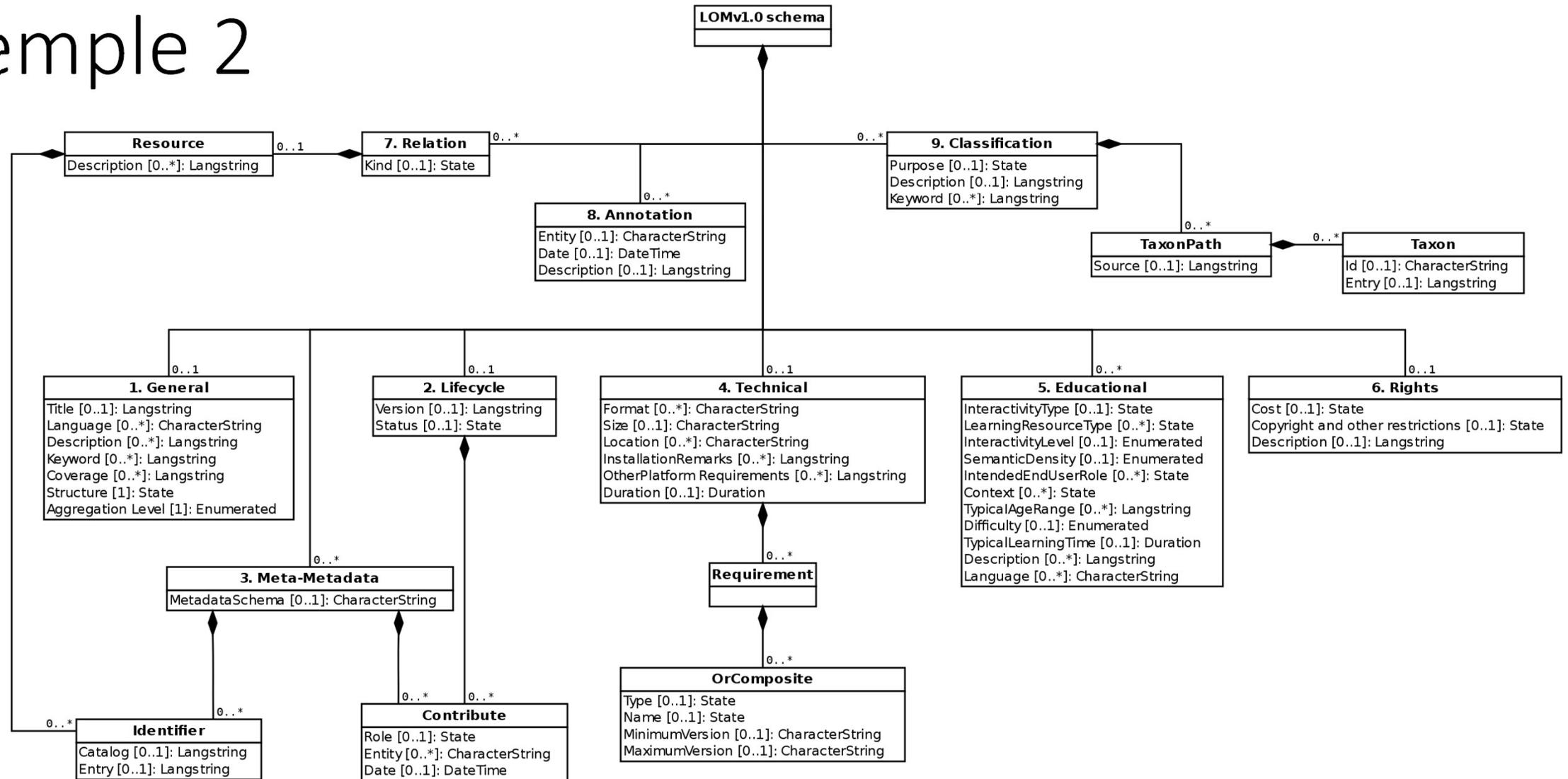


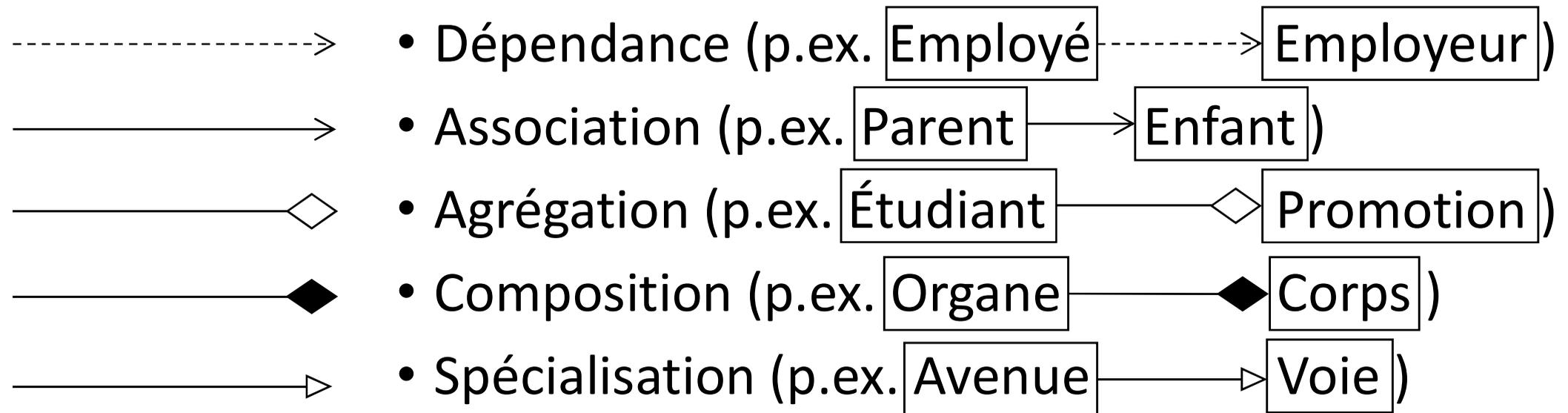
Diagramme de la hiérarchie
des classes de diagrammes de UML (trouvé sur Wikipédia, domaine public)

Exemple 2



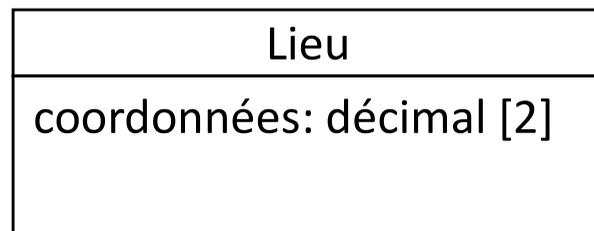
Source: wikipédia, de LOM_base_schema.svg créé par EleGale sous license CC-BY-SA 4.0

Les types de relations



Cardinalités

- 1 unicité
- 0..1 optionalité
- * aucune contrainte (0, 1 ou plusieurs)
- 1..* obligatoire et peut être multiple
- On peut également indiquer la multiplicité d'un attribut



Indique qu'il y a 2 valeurs
décimales pour cet attributs

Hiérarchie de classes

- On utilise la relation de spécialisation pour construire des hiérarchies permettant de classer les objets de façon plus ou moins spécifiques
 - Construire une hiérarchie des espaces d'un bâtiment (pièces, couloirs, etc.)
 - Construire une hiérarchie des éléments de construction (fenêtre, poutre, etc.)