Chapter 01 /Part 02: From Conceptual to Relational Model

Dr. Djakhdjakha L.

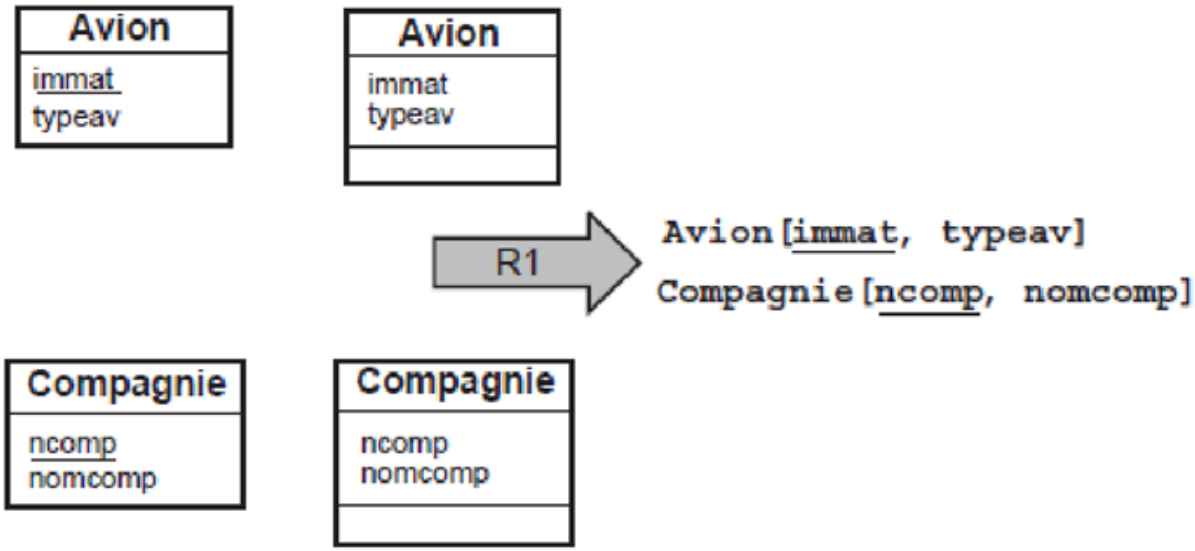
February 2025

From Conceptual to Logical

From an entity-relationship/UML schema to a relational schema.

Entity/Class Transformation

- **Rule 1:**
- Each entity becomes a relation.
- The entity identifier becomes the primary key of the relation.
- Each class in the UML diagram becomes a relation.



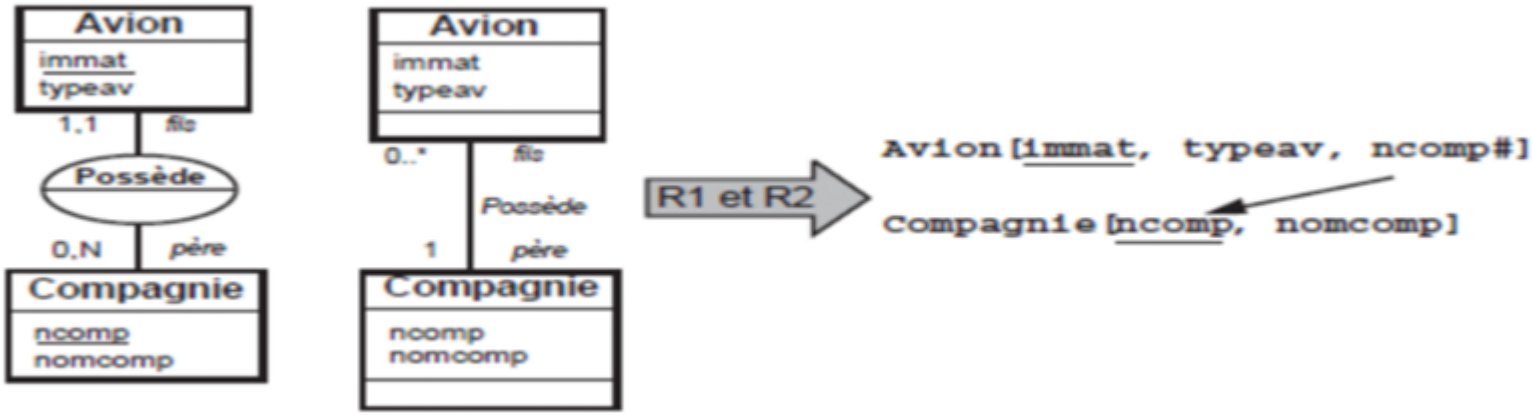
- Select an attribute of the class to act as an identifier.
- If no attribute is suitable, add one to ensure the relation has a primary key.

Association Transformation

- Transformation rules depend on the cardinality/multiplicity of associations:
- One-to-Many, Many-to-Many, Class-Associations, N-ary, One-to-One.

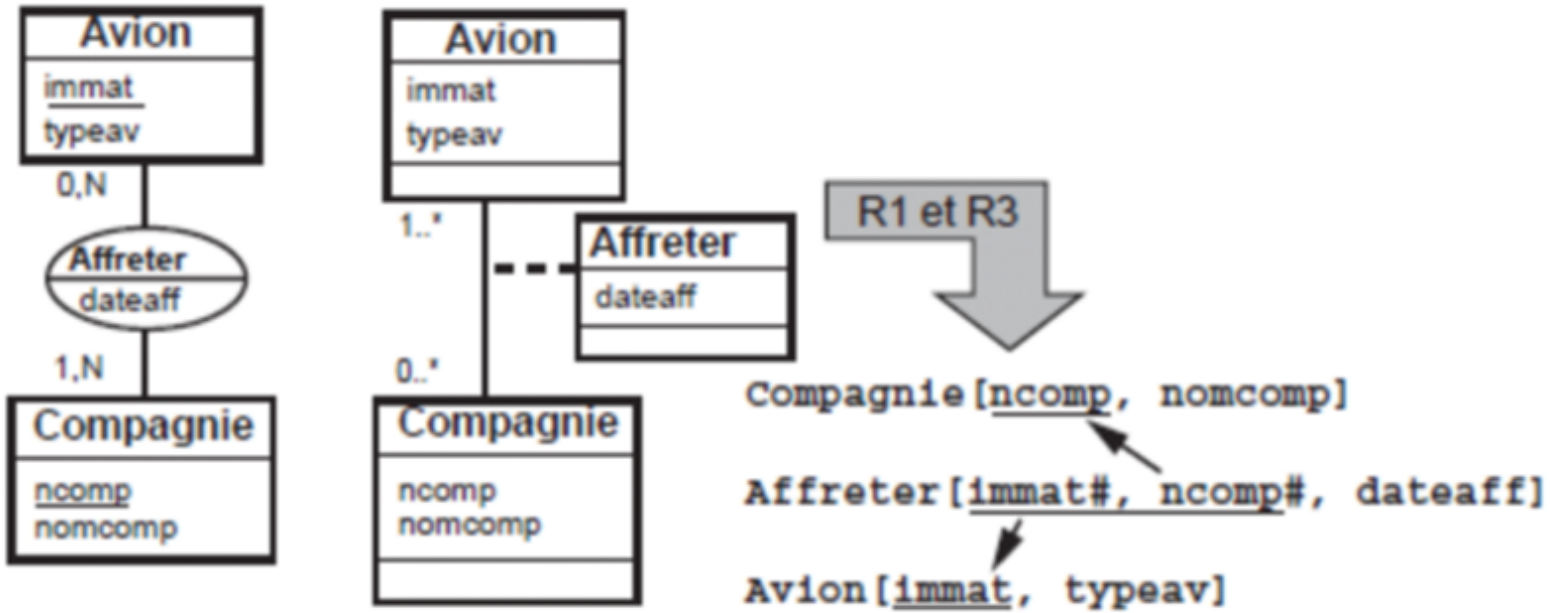
One-to-Many Associations

- **Rule 2:**
- Add a foreign key attribute in the child relation of the association.
- The attribute takes the name of the primary key of the parent relation of the association.

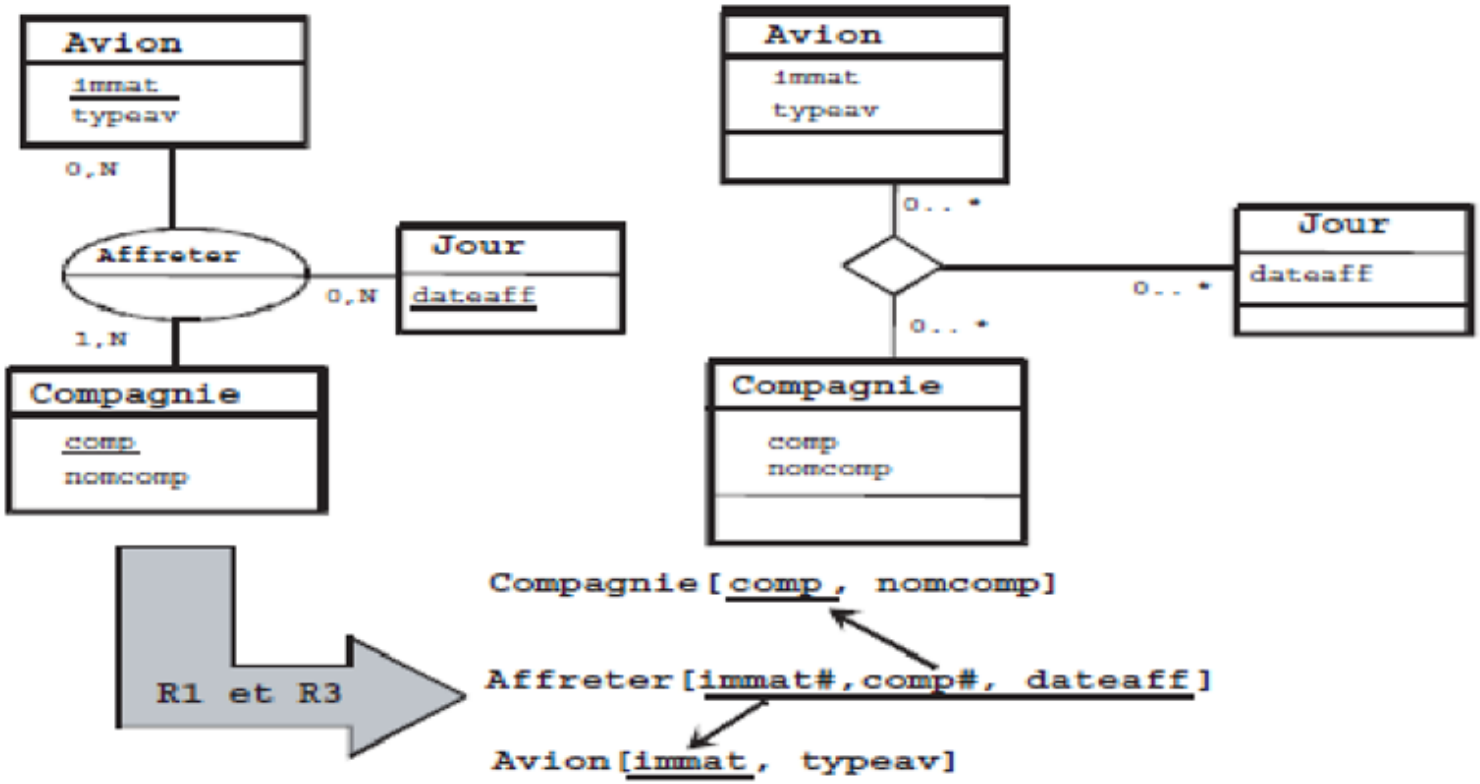


Many-to-Many and N-ary Associations

- **Rule 3:**
- The association (class-association) becomes a relation.
- The primary key is formed by concatenating the identifiers of the entities (classes) connected to the association.
- Attributes of the association should be added to the new relation but are neither primary nor foreign keys.

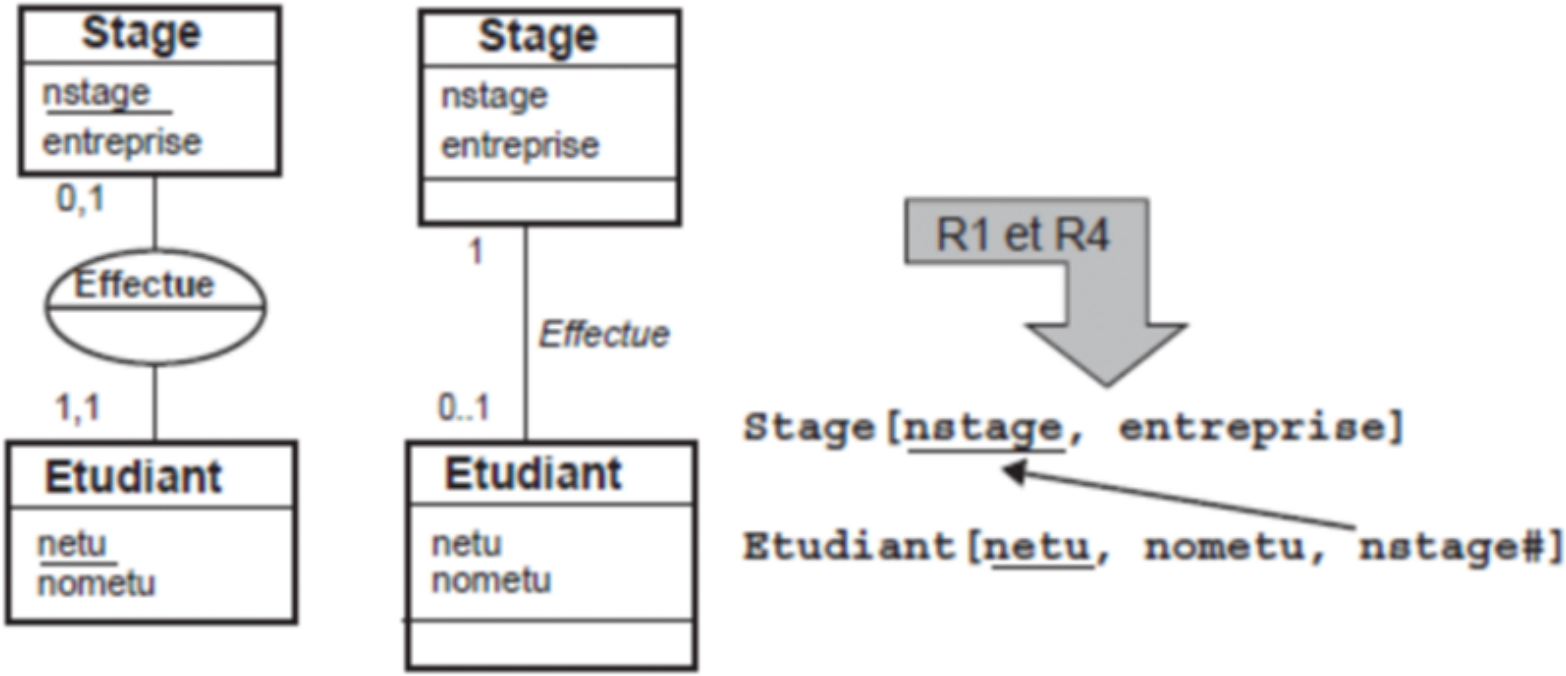


Many-to-Many and N-ary Associations



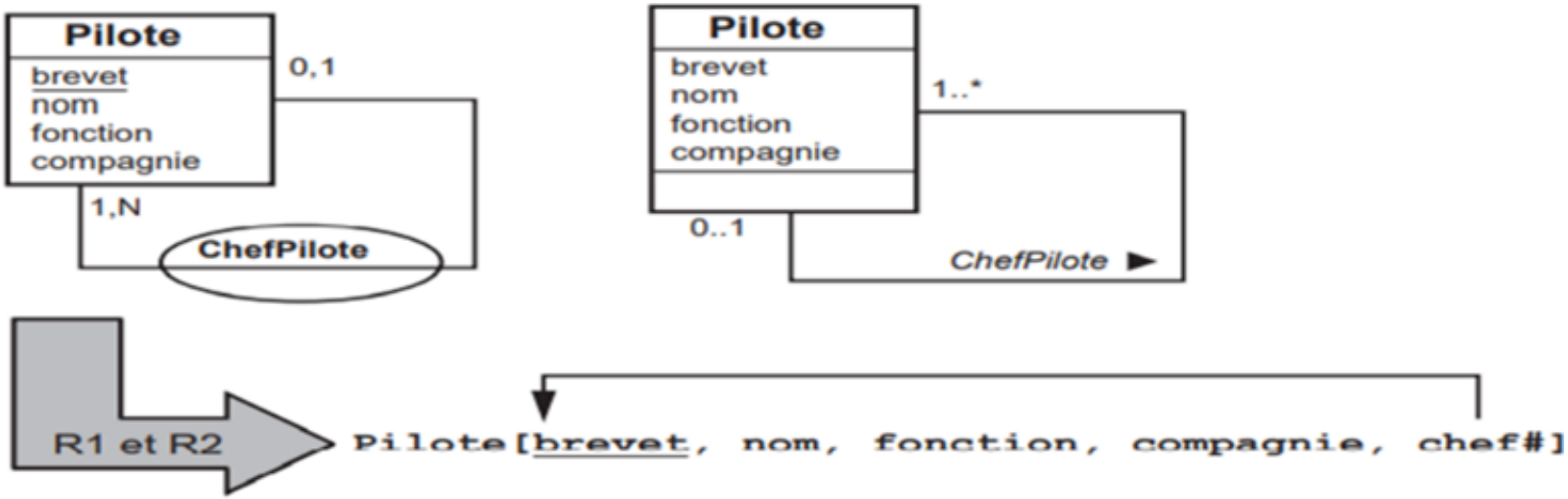
One-to-One Associations

- **Rule 4:**
- To avoid NULL values in the database, add a foreign key attribute in the relation derived from the entity with a minimum cardinality of zero.
- If both minimum cardinalities are zero, a choice is made between the two relations.
- If both are one, merging the two entities (classes) is recommended.



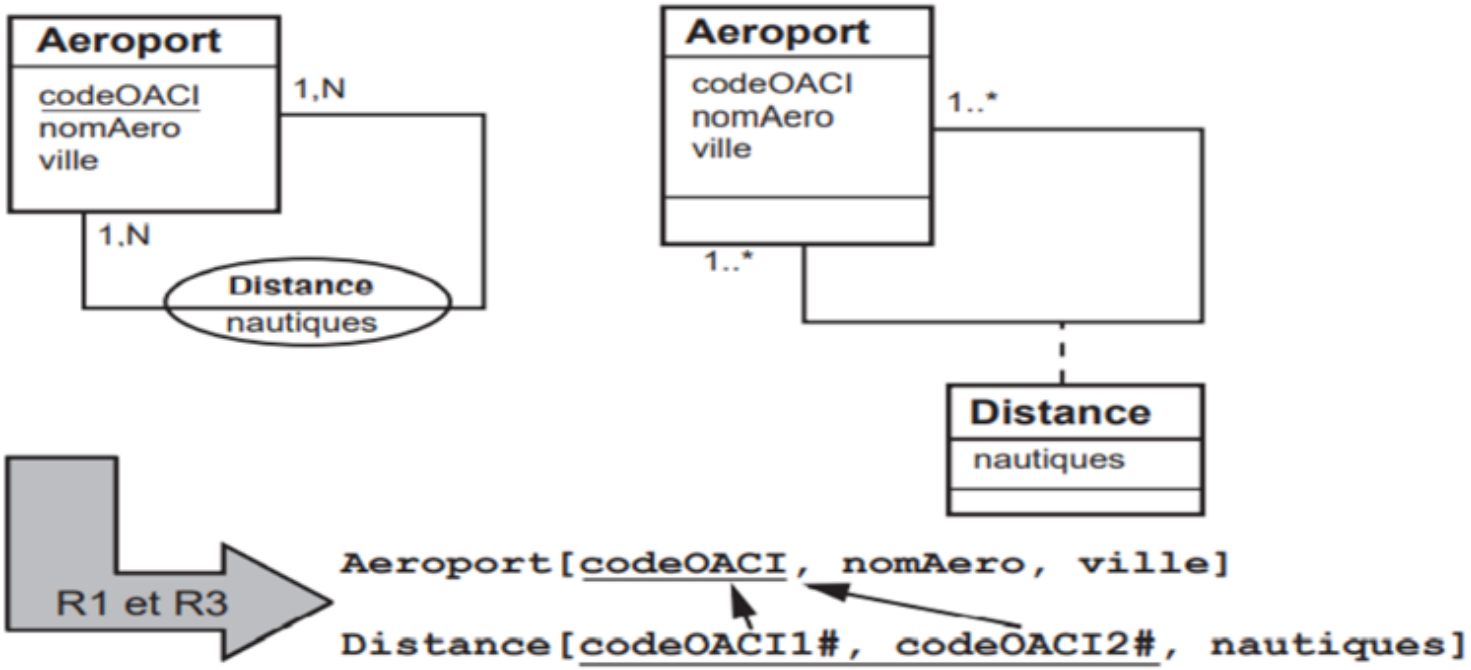
Recursive Associations

Recursive Associations One-to-Many



Recursive Associations

Recursive Associations Many-to-Many

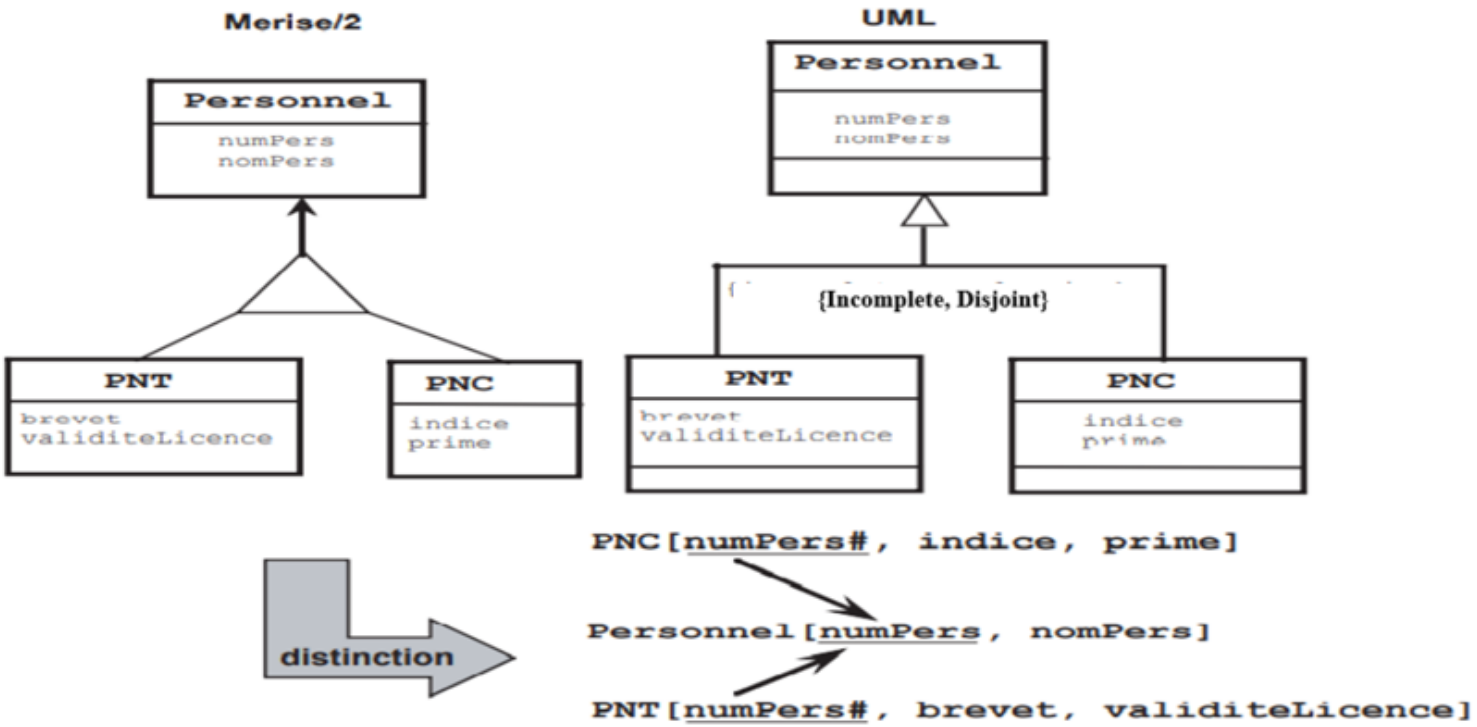


Inheritance Transformation

- Decomposition by Distinction
- Push-down Decomposition
- Push-up Decomposition

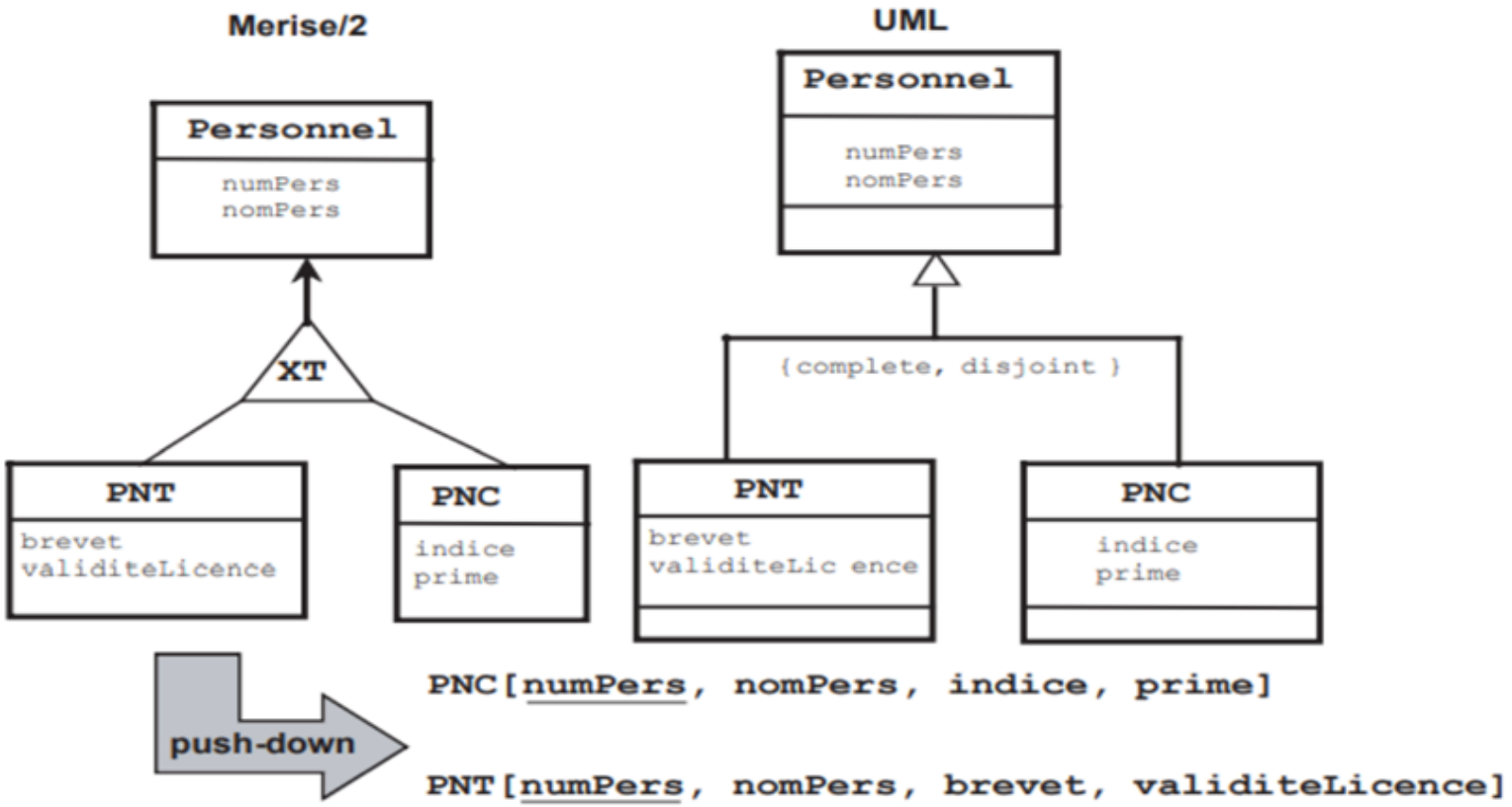
Decomposition by Distinction

- Transform the superclass into a relation.
- Transform each subclass into a relation.
- The primary key of the superclass migrates into the subclass relation(s) and becomes both a primary and foreign key.



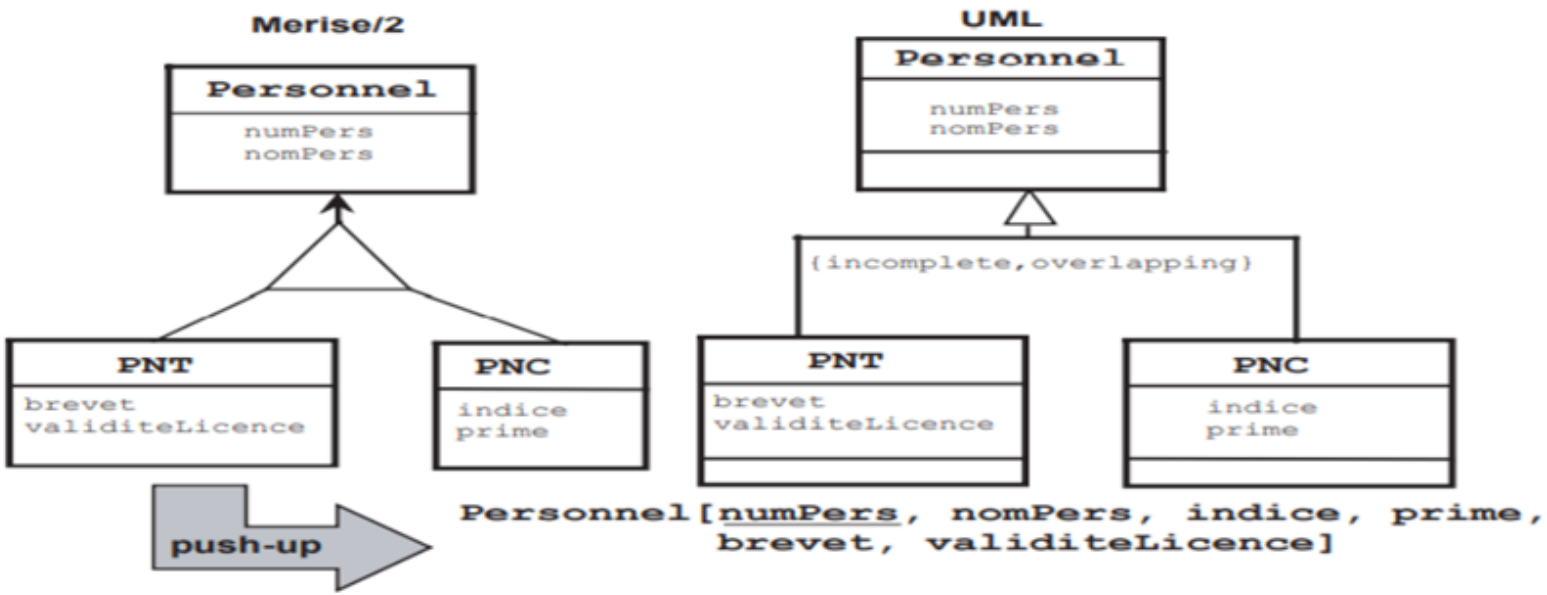
Push-down Decomposition

- If a totality or partition constraint exists, the superclass relation may not be translated.
- All attributes migrate into the subclass relation(s).

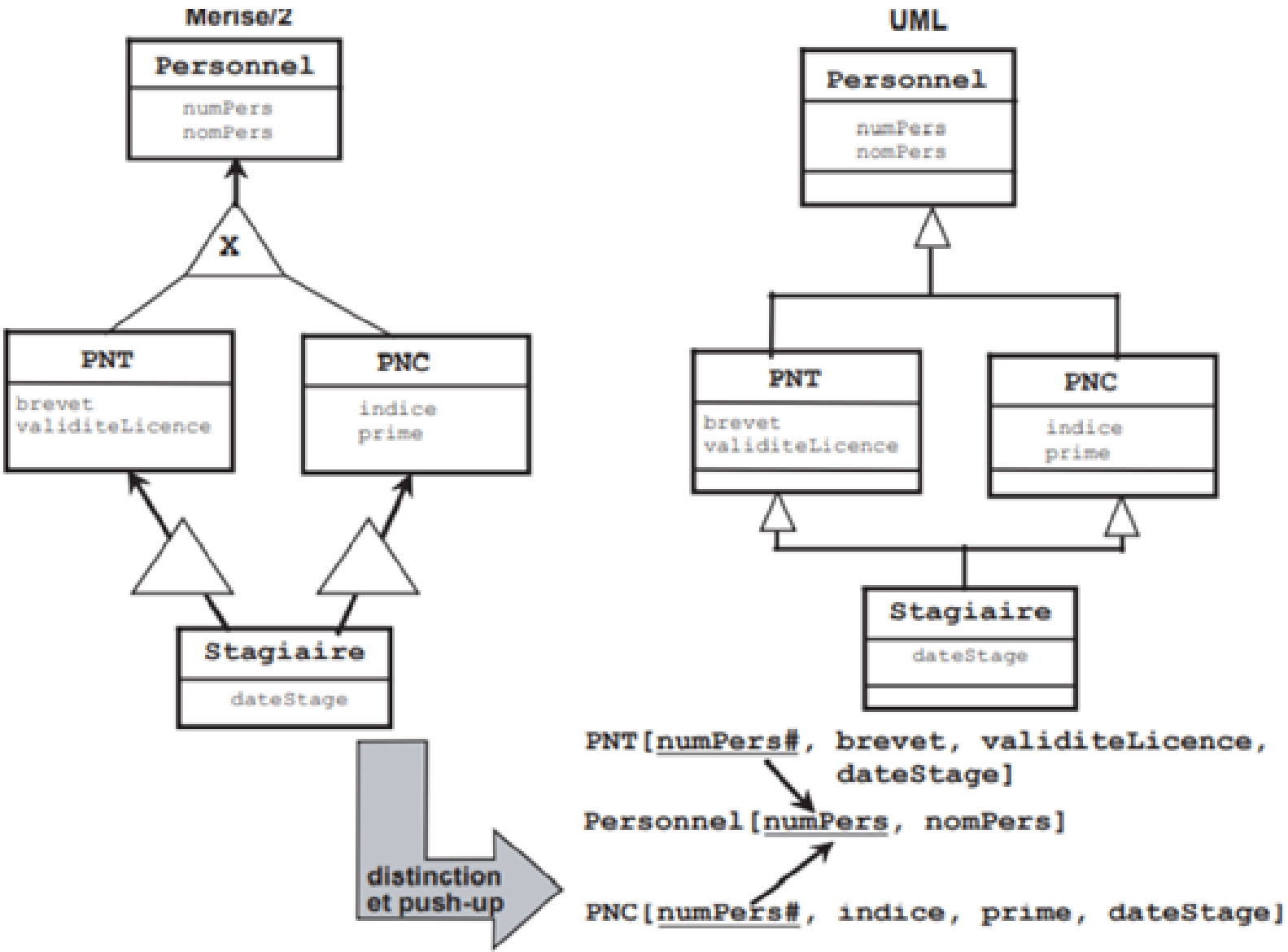


Push-up Decomposition

- Remove subclass relations and migrate their attributes into the superclass relation.

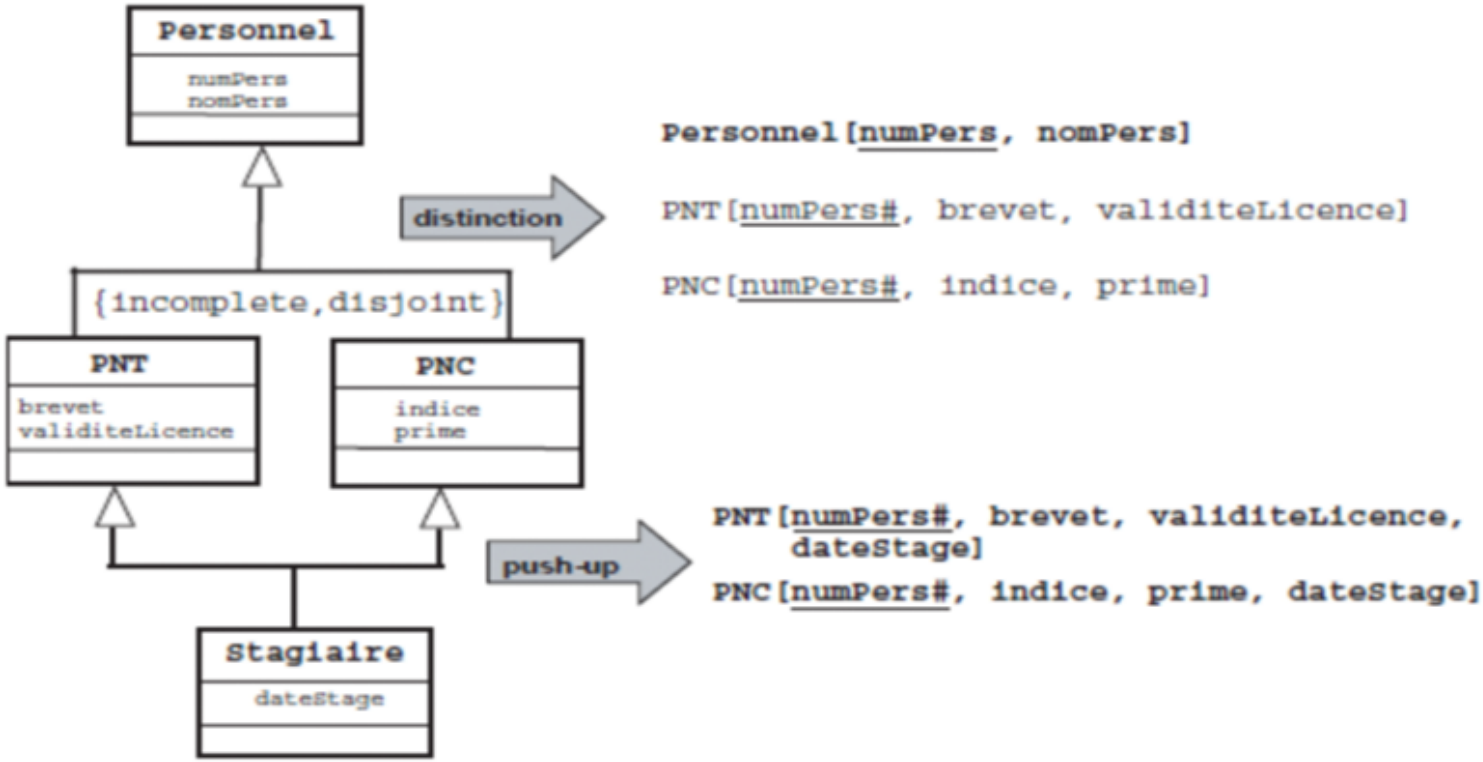


Multiple Inheritance



Multiple Inheritance

Transformation of multiple inheritance associations.



Composition Transformation

- The primary key of the relations derived from component classes must contain the identifier of the composite class, regardless of multiplicities.