

# CEBD - Projet (Partie 1)

## 1. Normalisation

### Question 1.

• **Communes (commune, departement, region, altitude\_moyenne, superficie, population, code\_commune, code\_canton, code\_arrondissement, code\_departement, code\_region)**

Les dépendances fonctionnelles sont :

- departement  $\rightarrow$  region, code\_region, code\_departement
- code\_departement  $\rightarrow$  departement, region, code\_region
- region  $\rightarrow$  code\_region
- code\_region  $\rightarrow$  region
- commune, departement  $\rightarrow$  altitude\_moyenne, superficie, population, code\_commune, code\_departement, region, code\_region, code\_arrondissement

L'attribut code\_canton n'est jamais en partie droite des dépendances, il fait donc partie de la clé.

La fermeture de commune et departement {commune, departement}<sup>+</sup> contient tous les attributs sauf code\_canton.

Ainsi, {commune, departement, code\_canton}<sup>+</sup> contient tous les attributs de la relation Communes. La clé de Communes est donc (commune, departement, code\_canton).

On retire la dépendance fonctionnelle code\_departement  $\rightarrow$  departement, region, code\_region pour ne pas avoir un attribut clé (departement) en partie droite d'une dépendance fonctionnelle.

La forme normale de Communes est 1NF.

Un attribut non-clé (region) dépend d'une partie de la clé (departement). Ainsi Communes n'est pas en 2NF.

On veut donc faire la décomposition en BCNF de Communes.

Les dépendances region  $\rightarrow$  code\_region et code\_region  $\rightarrow$  region empêchent la BCNF car ni l'attribut region ni code\_region ne sont pas des attributs-clés.

On crée donc une première relation :

R1 (region, code\_region) avec region  $\rightarrow$  code\_region, et où la clé est region.

On aurait également pu faire R1(code\_region, region) avec code\_region → region et où la clé est code\_region. R1 est en BCNF.

On crée une seconde relation :

R2 (commune, département, altitude\_moyenne, region, superficie, population, code\_commune, code\_arrondissement, code\_département, code\_canton)

avec comme dépendances fonctionnelles :

- département → region, code\_département
- commune, département → altitude\_moyenne, superficie, population, code\_commune, code\_département, code\_arrondissement

et où la clé est (commune, département, code\_canton).

R2 est en 1NF mais pas en 2NF car un attribut non clé (région) dépend d'une partie de la clé (département). Toutes les dépendances fonctionnelles empêchent la BCNF.

On fait donc la décomposition BCNF de R2 :

On a une première relation :

R3 (département, region, code\_département) avec département → region, code\_département, la clé est département, et R3 est en BCNF.

On a une deuxième relation :

R4 (commune, département, altitude\_moyenne, superficie, population, code\_commune, code\_canton, code\_arrondissement)

avec comme dépendance fonctionnelle :

commune, département → altitude\_moyenne, superficie, population, code\_commune, code\_arrondissement

et ayant pour clé (commune, département, code\_canton)

R4 est en 1NF, mais pas en 2NF car (commune, département) n'est qu'une partie de la clé, or la dépendance fonctionnelle devrait contenir la clé entière pour être en BCNF.

Ainsi, on fait une décomposition BCNF de R4 en créant deux relations R5 et R6 :

R5 (commune, département, altitude\_moyenne, superficie, population, code\_commune, code\_arrondissement)

avec commune, département → altitude\_moyenne, superficie, population, code\_commune, code\_arrondissement

et avec pour clé (commune, département)

R5 est en BCNF.

R6 (commune, département, code\_canton)

Il n'y a pas de dépendances fonctionnelles, et la clé est (commune, département, code\_canton). R6 est en BCNF.

Ainsi, Communes  $\equiv$  R1\*R3\*R5\*R6.

• **Zones climatiques (code\_departement, nom\_departement, zone\_climatique)**

Les dépendances fonctionnelles sont :

- code\_departement → zone\_climatique, nom\_departement
- nom\_departement → zone\_climatique, code\_departement

D'après ces dépendances fonctionnelles, on peut déduire que l'on a deux clés distinctes : soit code\_departement, soit nom\_departement.

On écrira la relation ainsi :

Zones climatiques (code\_departement, nom\_departement, zone\_climatique)

Cette relation est en BCNF.

• **Mesures (date\_obs, code\_insee\_departement, departement, tmin, tmax, tmoy)**

Les dépendances fonctionnelles sont :

- code\_insee\_departement → departement
- departement → code\_insee\_departement
- date\_obs, code\_insee\_departement → departement, tmin, tmax, tmoy
- date\_obs, departement → code\_insee\_departement, tmin, tmax, tmoy

On a deux clés : (date\_obs, code\_insee\_departement) et (date\_obs, departement)  
Mesures est en 2NF, mais pas en BCNF.

On veut ainsi faire la décomposition BCNF de Mesures.

Si on prend comme clé (date\_obs, code\_insee\_departement)

Les dépendances code\_insee\_departement → departement et departement → code\_insee\_departement empêchent la BCNF (la partie gauche des dépendances ne sont pas clés).

On produit deux relations :

R1 (code\_insee\_departement, departement) avec comme clé code\_insee\_departement, et comme dépendance code\_insee\_departement → departement. R1 est en BCNF.

R2 (code\_insee\_departement, date\_obs, tmin, tmax, tmoy) avec comme clé (code\_insee\_departement, date\_obs), et comme dépendance fonctionnelle code\_insee\_departement, date\_obs → tmin, tmax, tmoy. R2 est en BCNF.

Si on prend comme clé (date\_obs, departement), on obtient les relations

R1 (departement, code\_insee\_departement) avec departement → code\_insee\_departement, avec comme clé departement. R1 est en BCNF.

R2 (departement, date\_obs, tmin, tmax, tmoy) avec comme clé (departement, date\_obs) departement, date\_obs → tmin, tmax, tmoy. R2 est en BCNF.

Mesures ≡ R1\*R2

## 2. Conception UML et transformation en relationnel

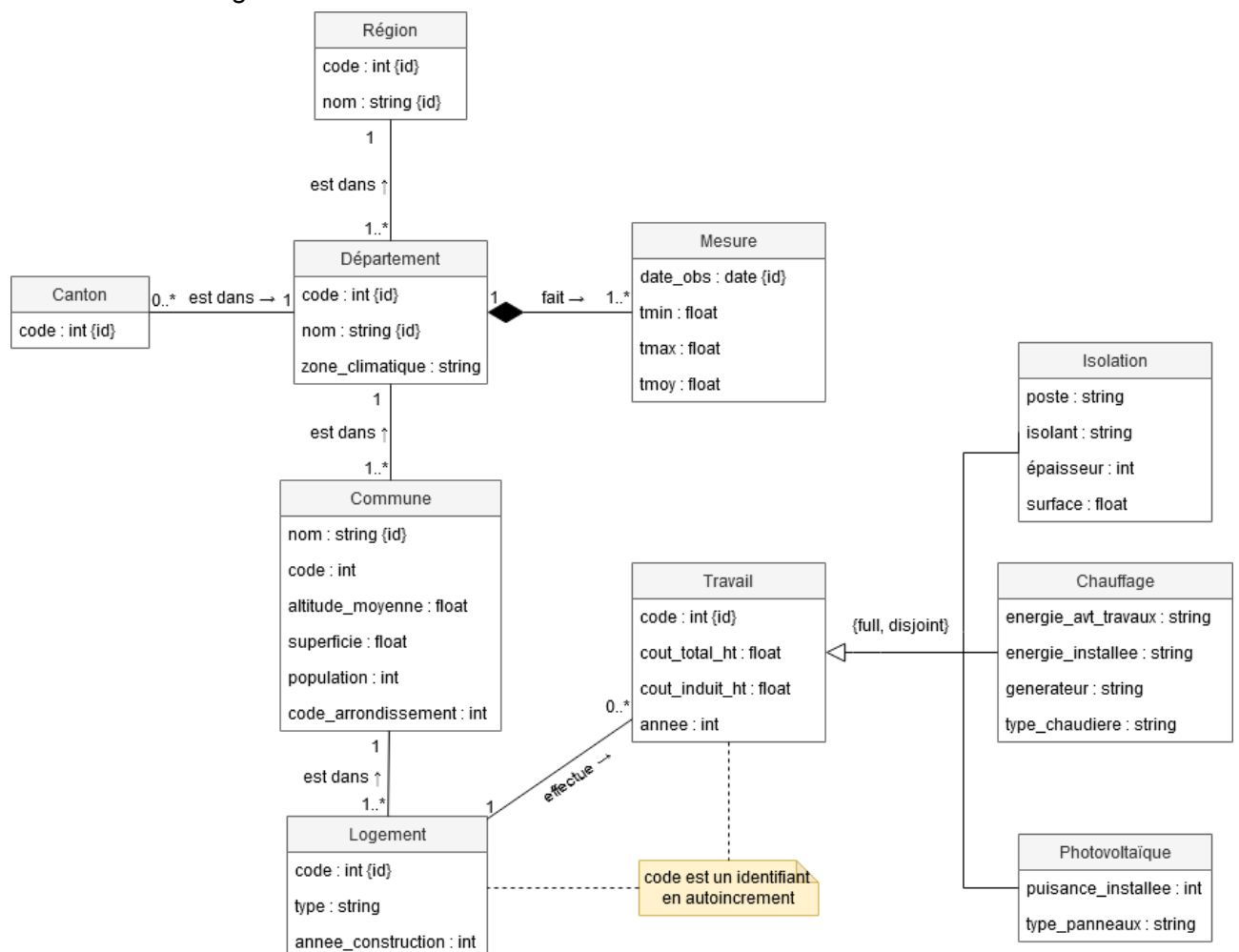
### Question 2.

- Pour les noms d'attributs, au niveau conceptuel (UML) on écrira uniquement le nom de l'attribut, tandis qu'au niveau logique (relationnel et SQL) on écrira l'attribut ainsi : attribut\_nomclasse

Par exemple, dans la relation Communes on a l'attribut population. Au niveau conceptuel on écrira *population* uniquement, et au niveau logique on écrira *population\_commune* (commune étant le nom de la classe UML).

- Pour les noms des relations, au niveau conceptuel (UML) on écrira le nom de la classe au singulier, tandis qu'au niveau logique (relationnel et SQL) on écrira le nom de la relation au pluriel.

### Question 3. Diagramme de classes UML



#### Question 4.

##### Modèle relationnel :

Cantons (code\_canton, code\_departement)

*/\* <c, d> ∈ Cantons ⇔ Le canton de code c est dans le département d. \*/*

Departements (code\_departement, nom\_departement, zone\_climatique\_departement, code\_region)

*/\* <c, n, z, r> ∈ Departements ⇔ Le département de code c est de nom n est une zone climatique z dans la région de code r. \*/*

Regions (code\_region, nom\_region)

*/\* <c, n> ∈ Regions ⇔ La région de nom n est de code c. \*/*

Communes (nom\_commune, code\_departement, code\_commune, altitude\_moyenne\_commune, superficie\_commune, population\_commune, code\_arrondissement\_commune)

*/\* <n, d, c, a, s, p, r> ∈ Communes ⇔ La commune de nom n du département de code d a pour code c, une altitude moyenne a, une superficie s, et une population p. Chaque code d'arrondissement de la commune est r. \*/*

Mesures (date\_obs\_mesure, code\_departement, tmin\_mesure, tmax\_mesure, tmoy\_mesure)

*/\* <d, c, i, a, o> ∈ Mesures ⇔ La mesure du département de code c à la date d a montré une température minimale i, une température maximale a et une température moyenne o. \*/*

Logements (code\_logement, type\_logement, annee\_construction\_logement, nom\_commune)

*/\* <l, t, a, c> ∈ Logements ⇔ Le logement de code l est de type t, a été construit l'année a, et est dans la commune de nom c. \*/*

Travaux (code\_travaux, cout\_total\_ht\_travaux, cout\_induit\_ht\_travaux, annee\_travaux, code\_logement)

*/\* <c, t, i, a, l> ∈ Travaux ⇔ Les travaux de code c ont un cout total ht t, un cout induit ht i, faits en l'an a, dans le logement de code l. \*/*

Isolations (code\_travaux, poste\_isolation, isolant\_isolation, epaisseur\_isolation, surface\_isolation)

*/\* <c, p, i, e, s> ∈ Isolations ⇔ L'isolation de code c est de poste p, son isolant est i, son épaisseur e, et sa surface s. \*/*

Chauffages (code\_travaux, energie\_avt\_travaux\_chauffage, energie\_installee\_chauffage, generateur\_chauffage, type\_chaudiere\_chauffage)

*/\* <c, e, i, g, h> ∈ Chauffages ⇔ Le chauffage de code c avait pour énergie e avant les travaux, et a pour énergie i après son installation. Le générateur est g et la chaudière est de type h. \*/*

Photovoltaïques (code\_travaux, puissance\_installée\_photovoltaïque,  
type\_panneaux\_photovoltaïque)

/\* <c, p, t> ∈ Photovoltaïques ⇔ Les panneaux photovoltaïques installés sont de type t,  
de puissance p et de code travaux c. \*/

### Domaines :

domaine (code\_canton) = domaine (code\_departement) = domaine (code\_commune) =  
domaine (population\_commune) = domaine (code\_arrondissement\_commune) = domaine  
(code\_region) = domaine (code\_travaux) = domaine (code\_logement) = domaine  
(annee\_travaux) = domaine (annee\_construction\_logement) = domaine  
(epaisseur\_isolation) = domaine (puissance\_installee\_photovoltaïque) = entiers > 0

domaine (nom\_region) = domaine (nom\_departement) = domaine (nom\_commune) =  
domaine (zone\_climatique\_departement) = domaine (type\_logement) = domaine  
(poste\_isolation) = domaine (isolant\_isolation) = domaine (energie\_avt\_travaux\_chauffage)  
= domaine (energie\_installee\_chauffage) = domaine (generateur\_chauffage) = domaine  
(type\_chaudiere\_chauffage) = domaine (type\_panneaux\_photovoltaïque) = chaîne de  
caractères

domaine (altitude\_moyenne\_commune) = domaine (superficie\_commune) = domaine  
(surface\_isolation) = réels > 0

domaine (tmin\_mesure) = domaine (tmax\_mesure) = domaine (tmoy\_mesure) = domaine  
(cout\_total\_ht\_travaux) = domaine (cout\_induit\_ht\_travaux) = réels

domaine (date\_obs\_mesure) = date

### Contraintes d'intégrité référentielles :

Departements[code\_region] ⊆ Regions[code\_region]

Cantons[code\_departement] ⊆ Departements[code\_departement]

Communes[code\_departement] ⊆ Departements[code\_departement]

Mesures[code\_departement] ⊆ Departements[code\_departement]

Logements[nom\_commune] ⊆ Communes[nom\_commune]

Travaux[code\_logement] ⊆ Logements[code\_logement]

Isolations[code\_travaux] ⊆ Travaux[code\_travaux]

Chauffages[code\_travaux] ⊆ Travaux[code\_travaux]

Photovoltaïques[code\_travaux] ⊆ Travaux[code\_travaux]

**Autres contraintes :**

Un travail est soit une installation d'isolation, soit de chauffage, soit de panneaux photovoltaïques.

C'est-à-dire que chaque code\_travaux distinct de la relation Travaux est soit dans la relation Isolations, soit dans Chauffages, soit dans Isolations, mais jamais dans deux ou trois relations héréditaires à la fois.