

UML

Unified Modeling Language



www.uml.org



www.omg.org

UML

DIAGRAMME de CLASSES

Diagramme de classe

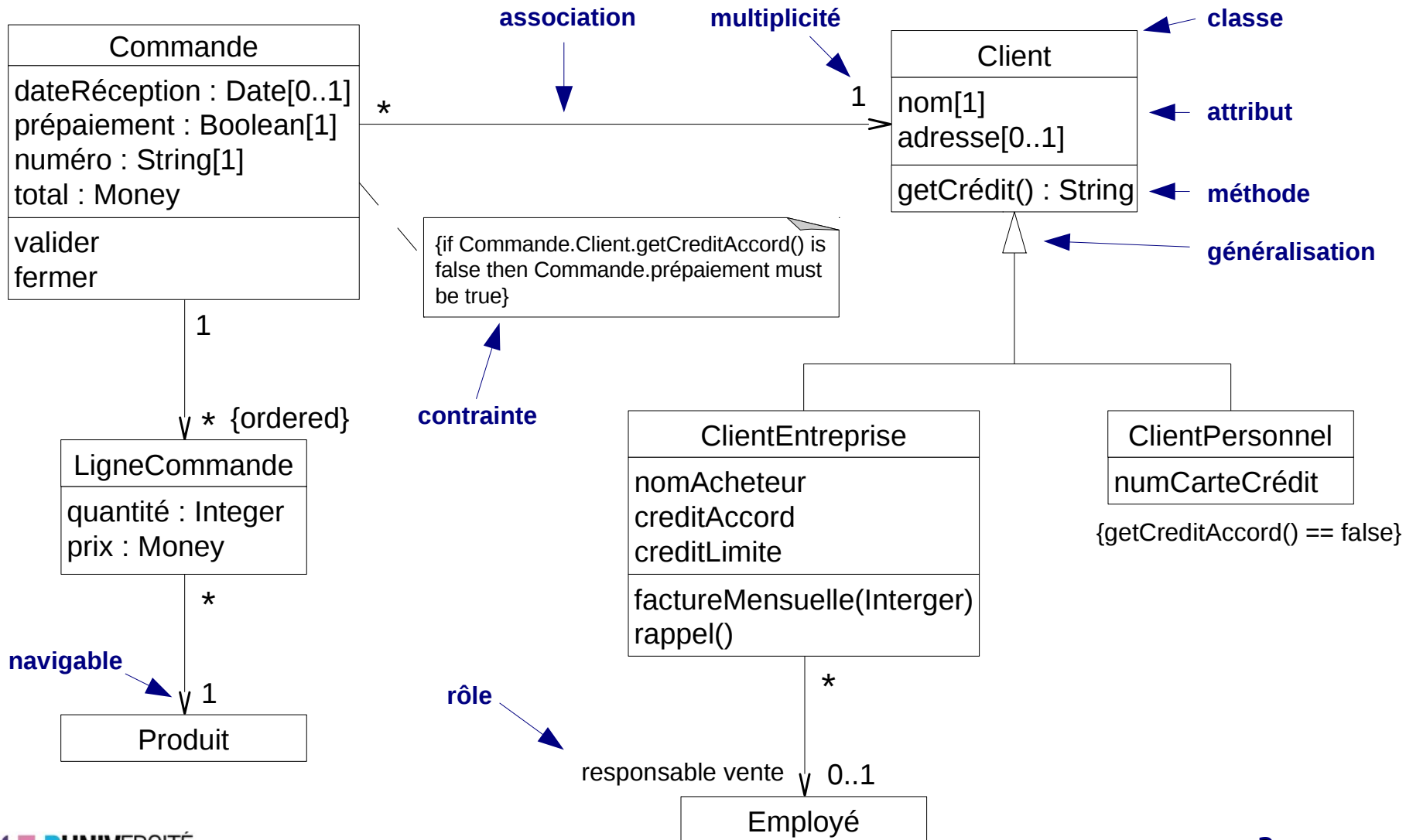


Diagramme de classe

Exemple de classe :

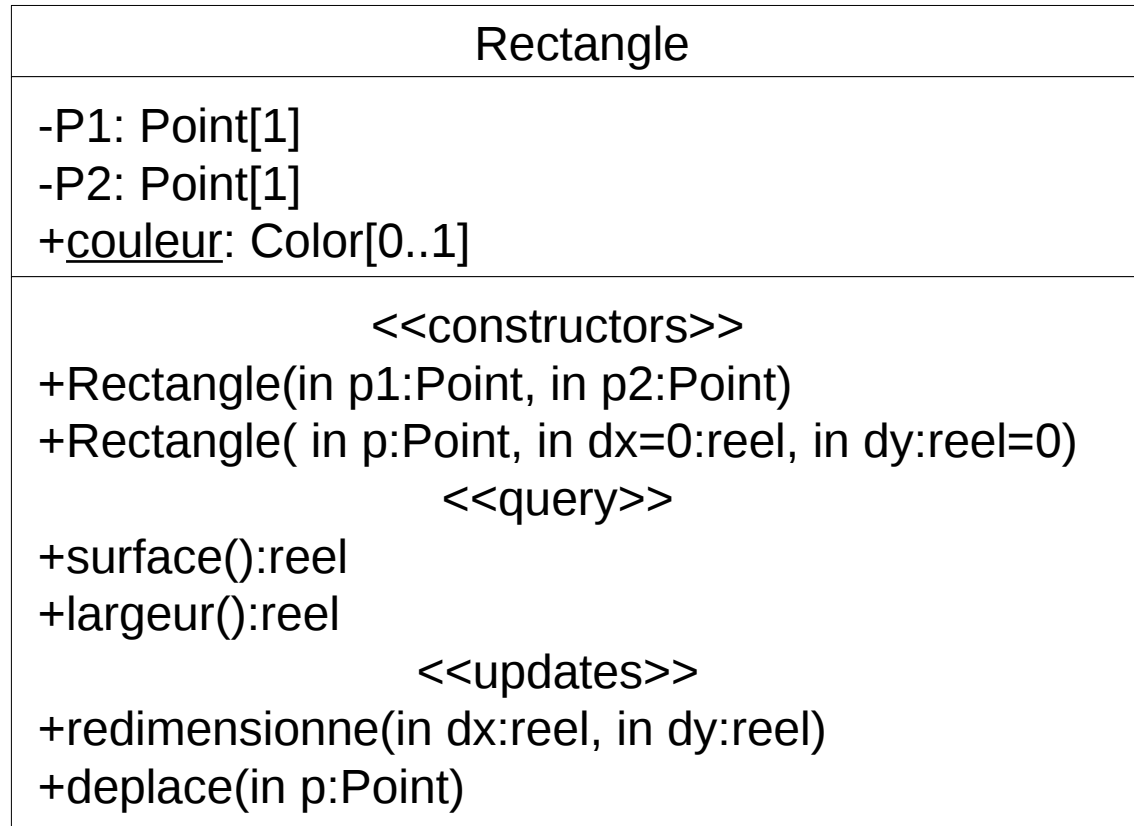


Diagramme de classe

Attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

- a) Conceptualisation
=> Notation UML2 indépendante du langage
- b) Pour générer le code
=> Notation UML en adéquation avec langage
- c) Phase étude : diagramme préliminaire
=> Toutes les informations ne sont pas définies

Diagramme de classe

Attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

visibilité	+	Public	-	Privé
	#	Protégé	~	Package

! Définition dans UML2
Variations selon langage utilisé

nom *soulignement* → *statique*

Diagramme de classe

Attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

type Restriction sur le format

! Formats défini dans UML

null

String | Boolean | Integer | Real | unlimitedNatural

Date

Diagramme de classe

Multiplicité des attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

attribut optionnel 0

attribut obligatoire borne mini ≥ 1

attribut monovalué borne sup = 1

attribut multivalué borne mini > 1 ou *

Remarque : Pas de multiplicité discontinue dans UML 2

Diagramme de classe

Multiplicité des attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

Expression	Structure d'arbre feuille / opérande	x xor b plus(x,1)
String expression	Chaîne de caractères	
Opaque expression	Langage externe	{OCL} i > j and self.size > i average hours worked per week

Diagramme de classe

Multiplicité des attributs :

visibilité nom : type[multiplicité] = valeur {contrainte}

contrainte

Mot-clé

OCL

Langage externe

ex : - nom : String [1] = "Durand" {readOnly}

Diagramme de classe

Exemple de contraintes :

Mots clé :	{ordered}	liste ordonnée
	{noUnique}	doublons autorisés
	{readOnly}	lecture seule
	{frozen}	"gelé"
	{notEmpty}	non vide
	{subset}	sous-ensemble
	{union}	regroupement
	...	

Diagramme de classe

Expression graphique de la contrainte :

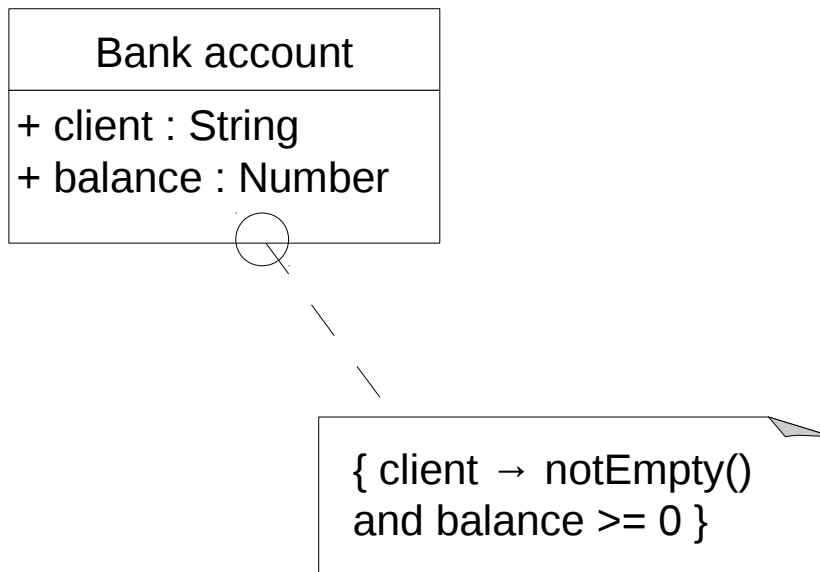
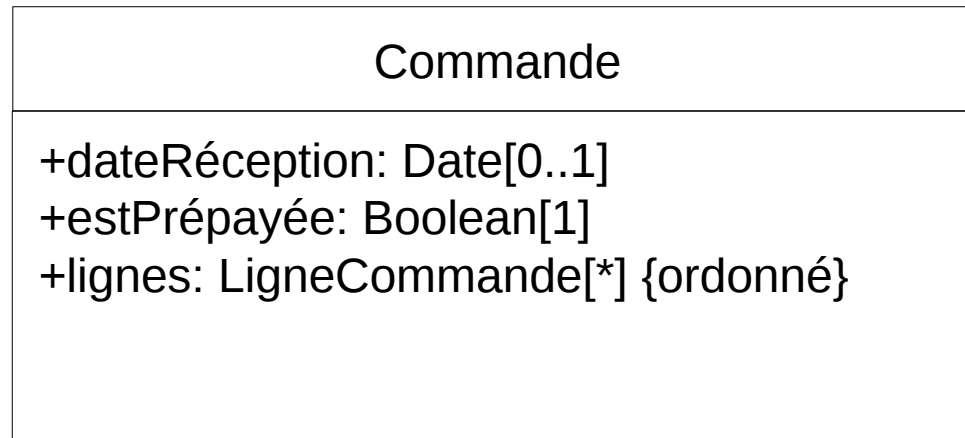


Diagramme de classe

Date : type primitif ?



ou association ?

Diagramme de classe

Opérations :

visibilité nom (arguments) : type retour {propriété}

visibilité +|-|#|~ public / privé ...

nom identifiant (chaîne car.)

arguments liste des paramètres

direction nom : type = défaut

direction : in, out, inout

défaut : valeur par défaut

retour format

propriété propriétés supplémentaires

Diagramme de classe

Association bidirectionnelle :

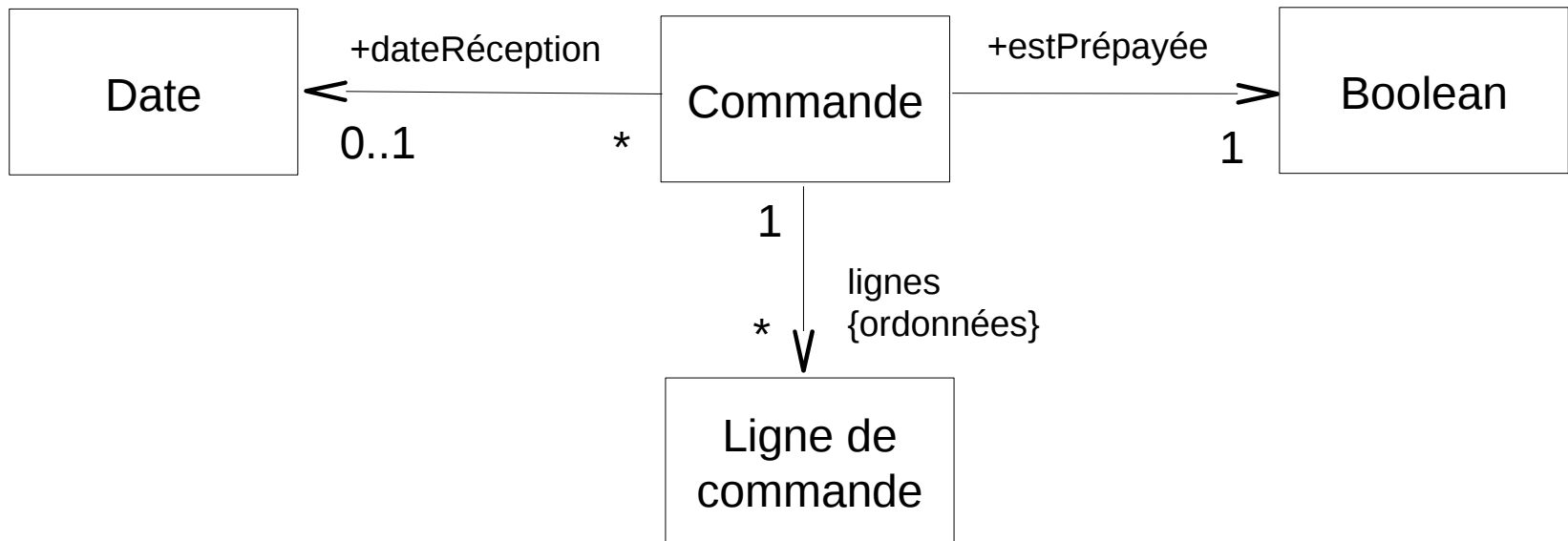


Par défaut : une association est bidirectionnelle



Diagramme de classe

Association :



Précise un sens de navigabilité

Diagramme de classe

Navigabilité uni-directionnelle :



Notations équivalentes

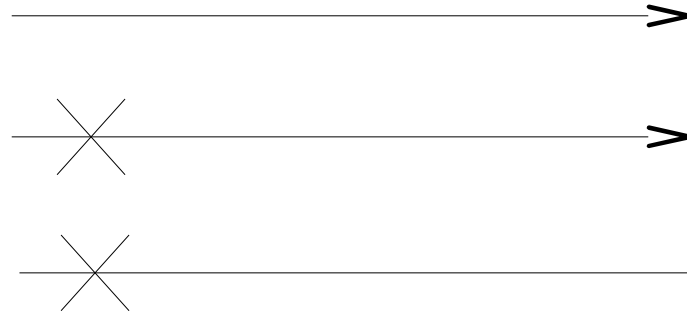


Diagramme de classe

Expression des multiplicités :

valeur stricte 1..1 ou 1

0 ou plus 0..* ou *

plus de 1 1..*

Diagramme de classe

Expression des multiplicités :

contrainte	isOrdered	isUnique
{set}	false	true
{orderedSet}	true	true
{bag}	false	false
{sequence}	true	false

Diagramme de classe

Association :

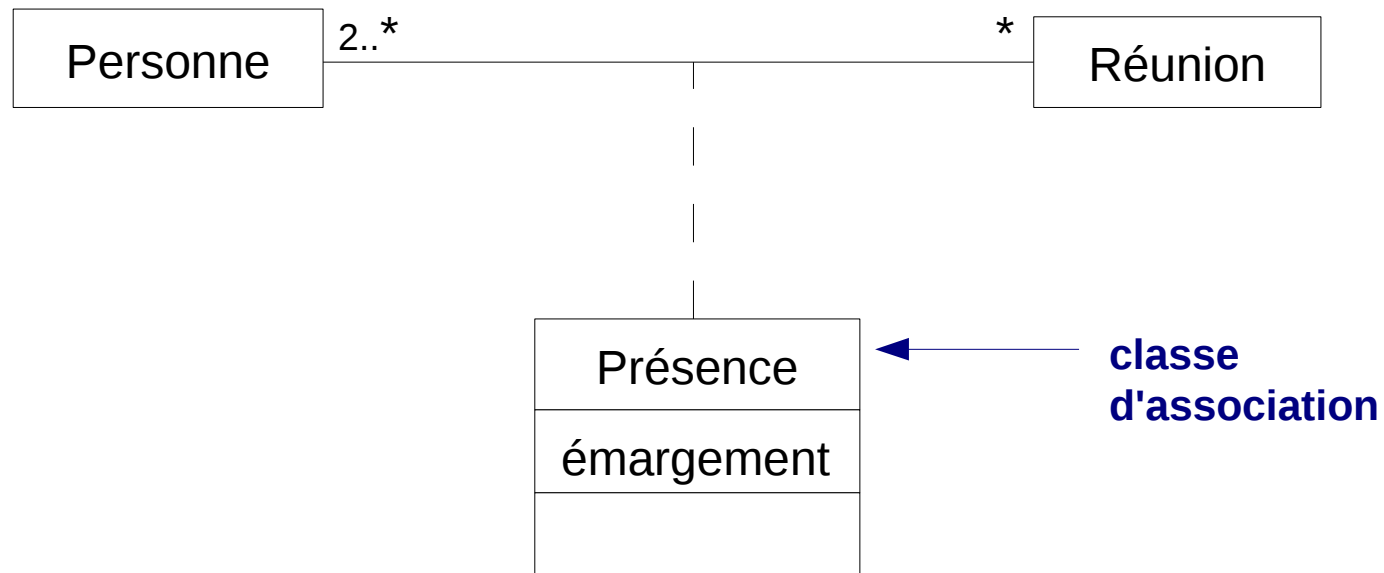
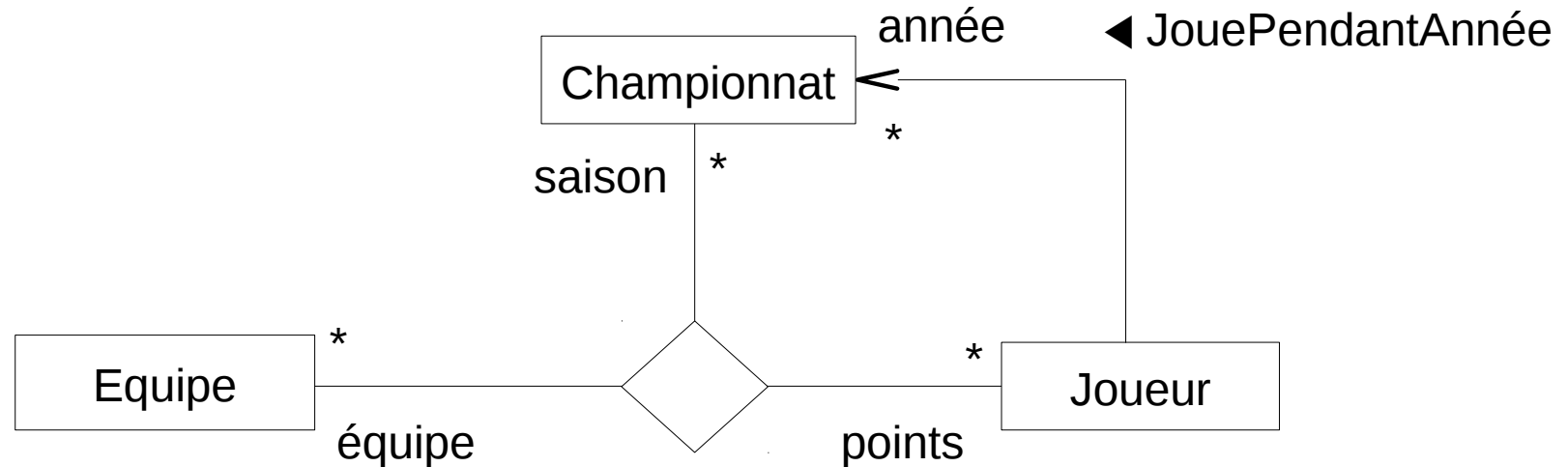


Diagramme de classe

Association n-aire :



Mutiplicité :

Si =1, alors toutes les combinaisons
ont au moins 1 relation avec élément !

! notation inversée des multiplicités

Diagramme de classe

Association – Equivalences de notation :

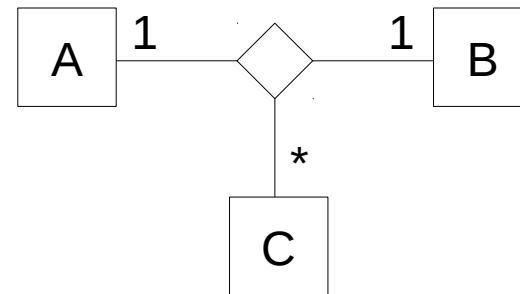
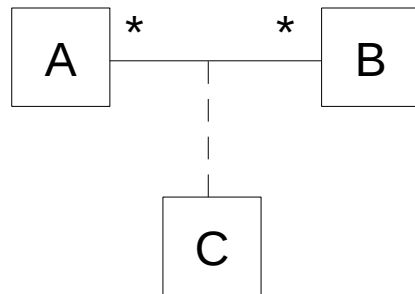
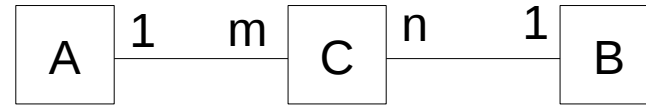
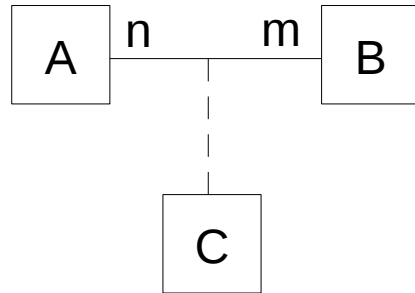


Diagramme de classe

Association qualifiée :



Qualificateur : Permet de restreindre la multiplicité de l'association

Diagramme de classe

Généralisation :

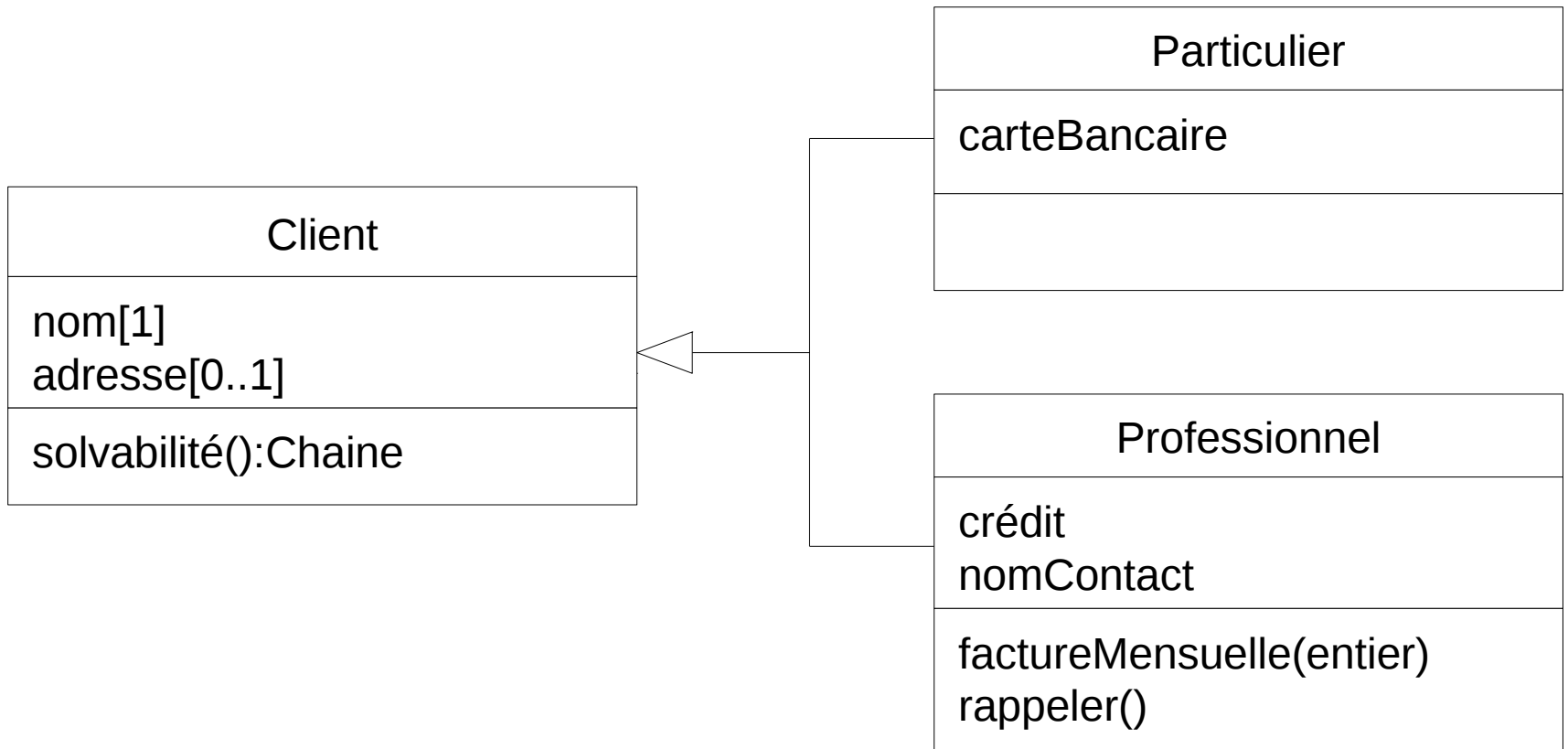
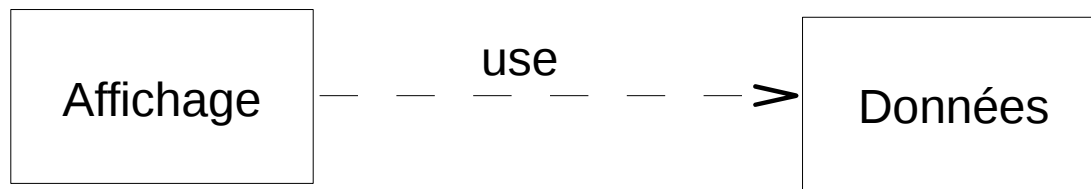


Diagramme de classe

Dépendance :



Mots clés

call

create

derive

instantiate

permit

realize

refine

substitute

trace

use

Diagramme de classe

Dépendance :

call	Appelle une opération de cible
create	Crée une instance de cible
derive	Source est dérivé de cible
instantiate	Source est une instance de cible
permit	Cible autorise l'accès privé à source
realize	Implémente une spécification ou interface de cible
refine	Relation entre niveaux sémantiques distincts
substitute	Source peut se substituer à cible
trace	Réalise un suivi d'exécution de cible (track)
use	Cible est requis pour implémenter source

Diagramme de classe

Contrainte :

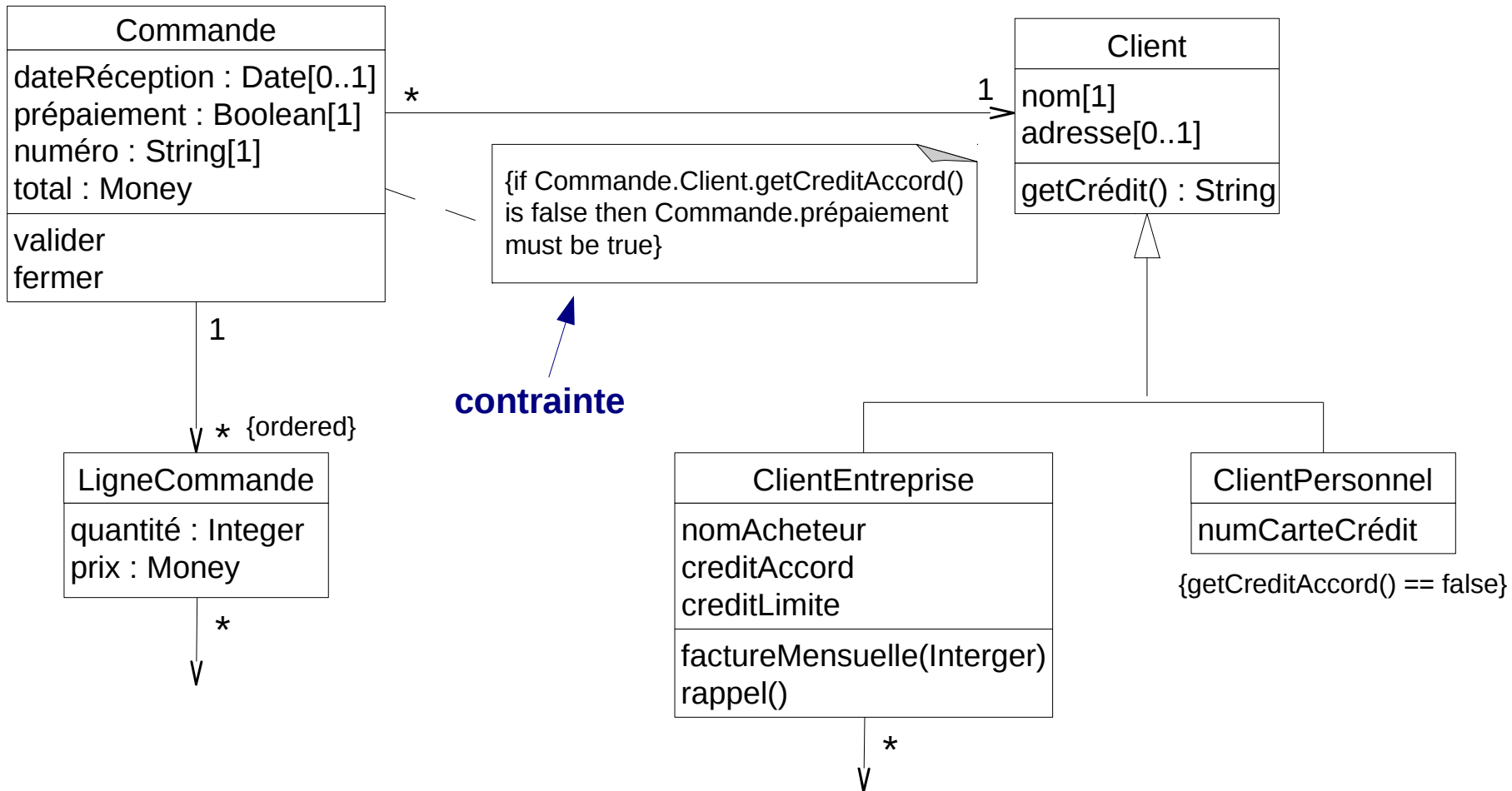


Diagramme de classe

Responsabilités :

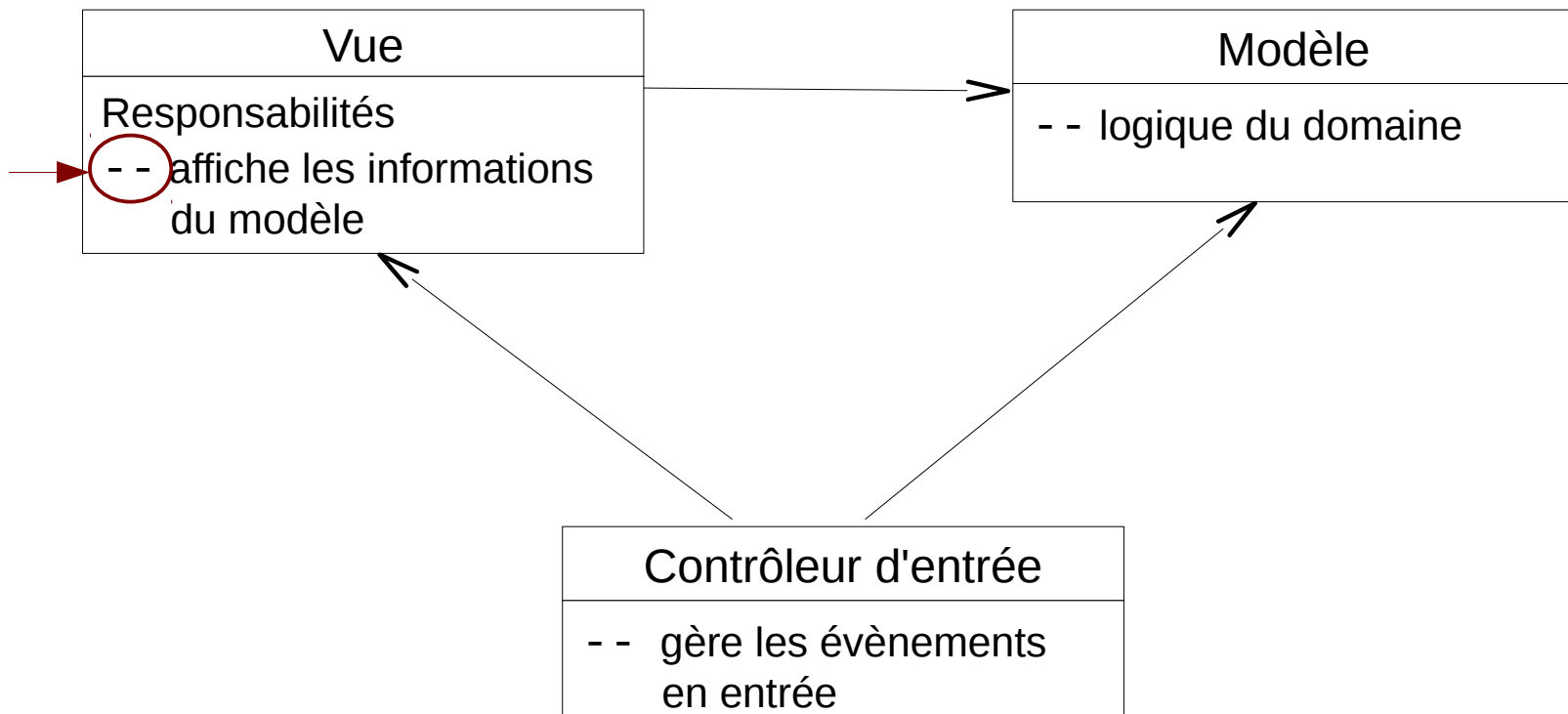
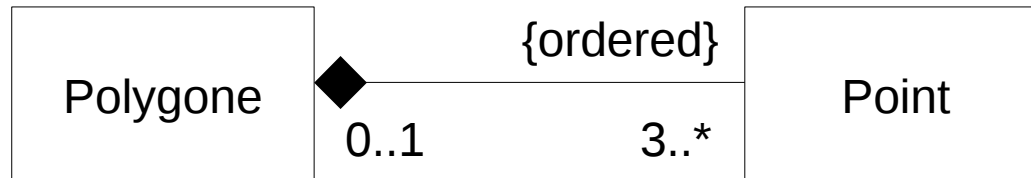


Diagramme de classe

Relation de composition :



Relation d'agrégation :



Diagramme de classe

Agrégation ou Composition



La suppression A entraine la suppression de B ? → C

B est-il utilisé en dehors de A ? → A

A et B sont-ils indépendants ? → A

A : Agrégation

C : Composition

Diagramme de classe

Propriété dérivée :

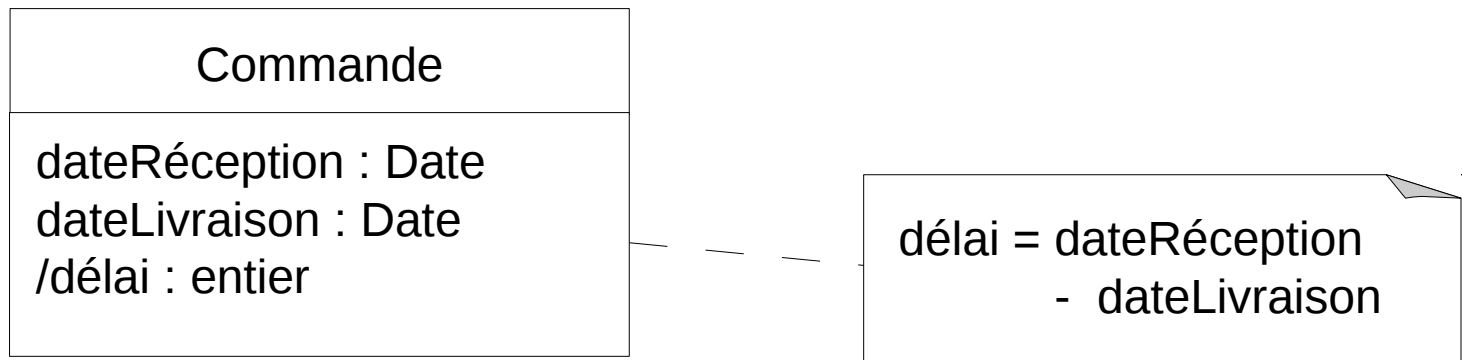


Diagramme de classe

Interface et classe abstraite :

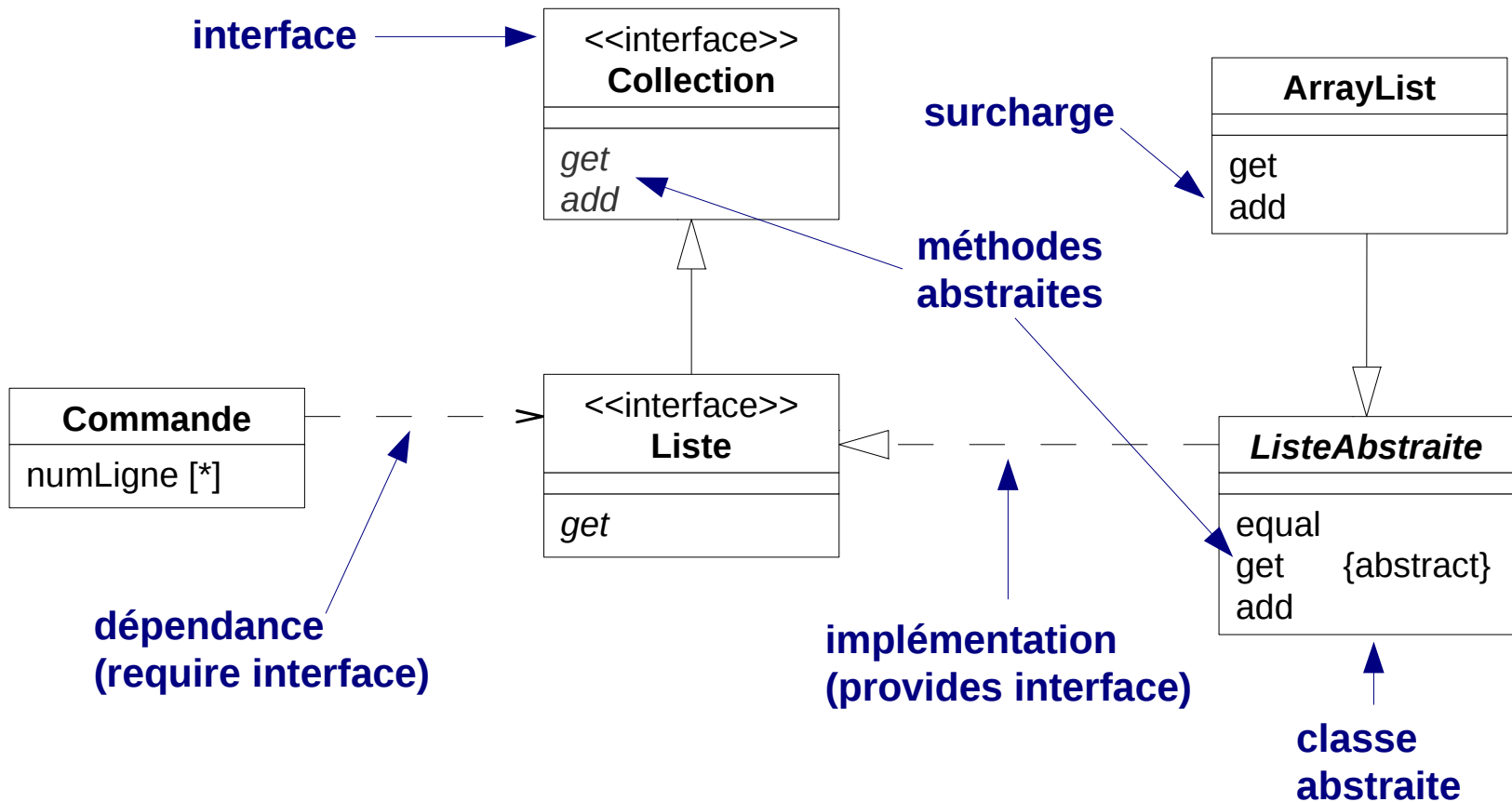


Diagramme de classe

Notation des interfaces :

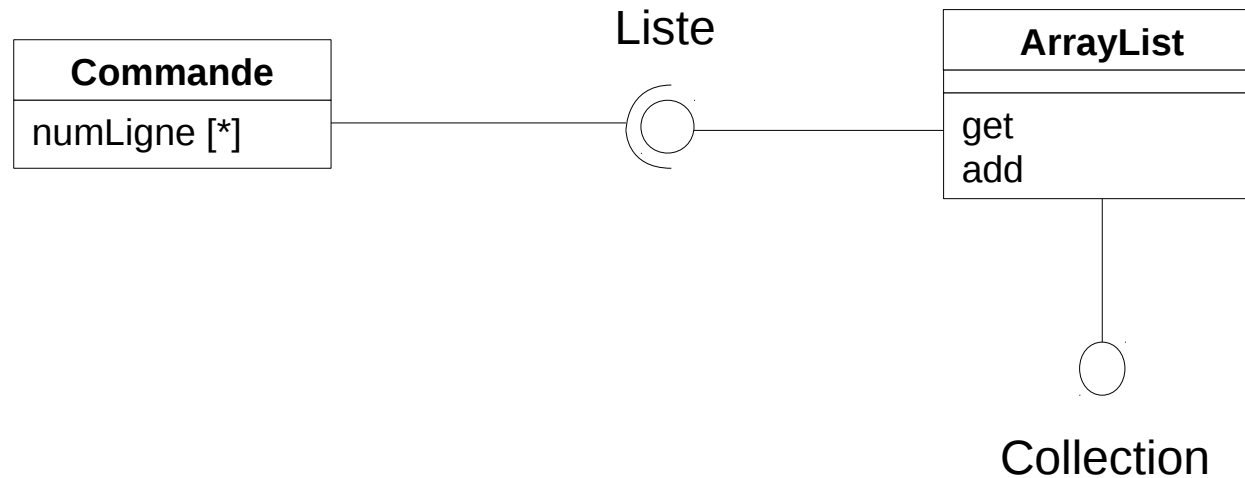


Diagramme de classe

Objet	Exemple	Comparaison
référence	Client	par référence objet
valeur	Date	des attributs

type de données << dataType >>

mot clé << value >> << struct >>

Diagramme de classe

Classification : Généralisation / Spécialisation

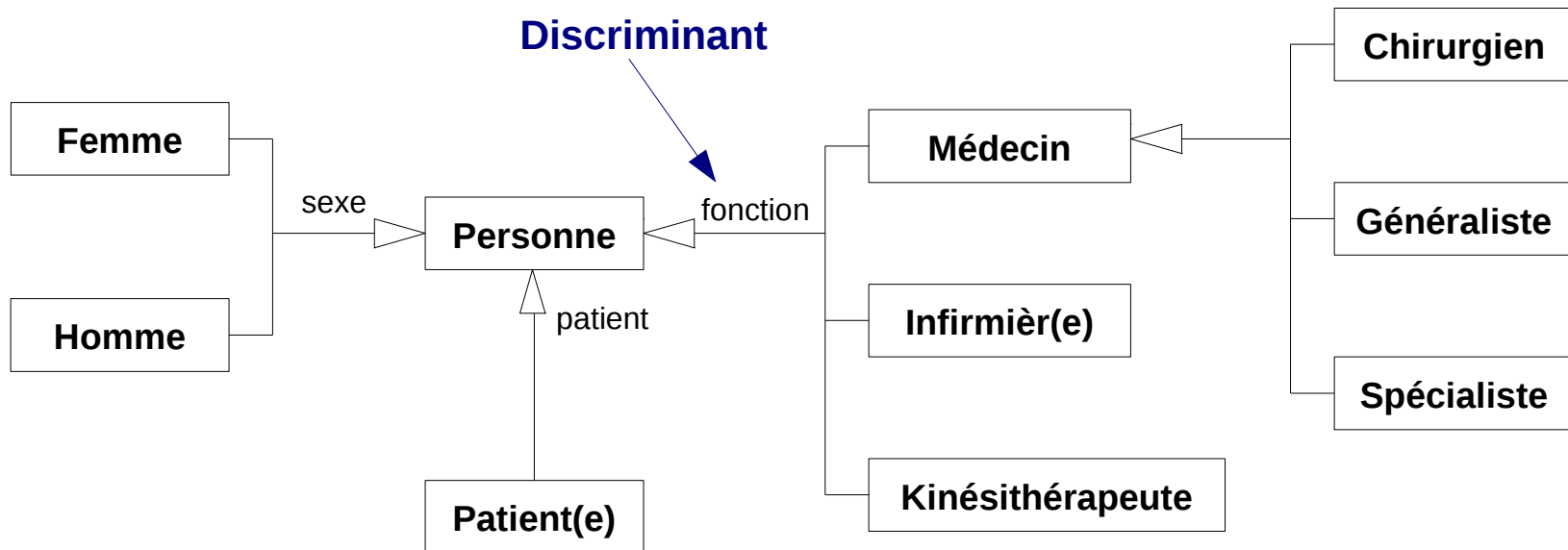
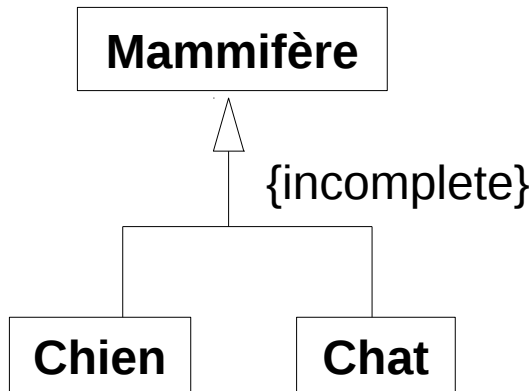


Diagramme de classe

Contrainte de classification :



Contrainte de complétude

{ complete } ↔ { incomplete }

Contrainte de chevauchement

{ disjoint } ↔ { overlapping }

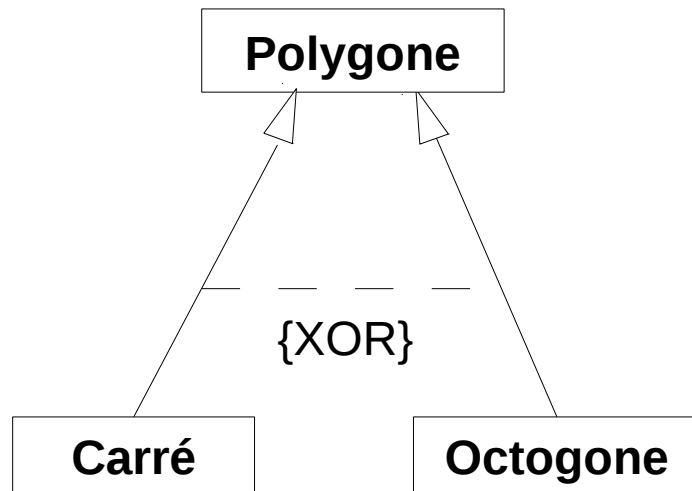
Combinaisons :

{ complete, disjoint }

...

Diagramme de classe

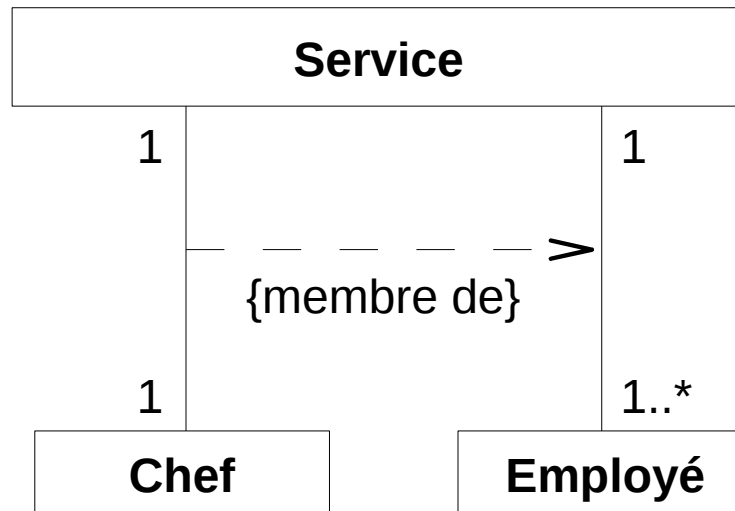
Association



Contrainte sur relation
OR
XOR
AND

Diagramme de classe

Association



Contrainte sur relation

Diagramme de classe

Classe paramétrable :

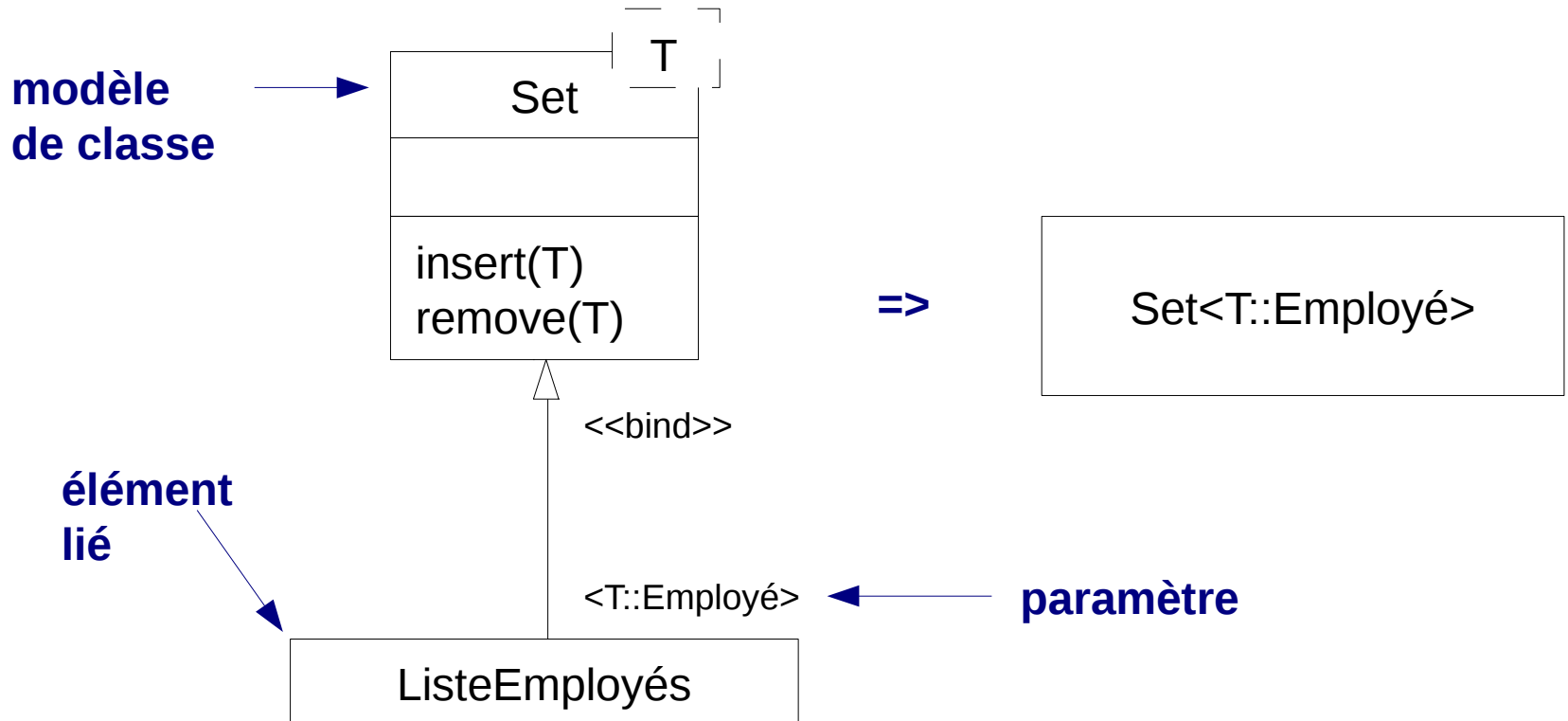


Diagramme de classe

Enumération :

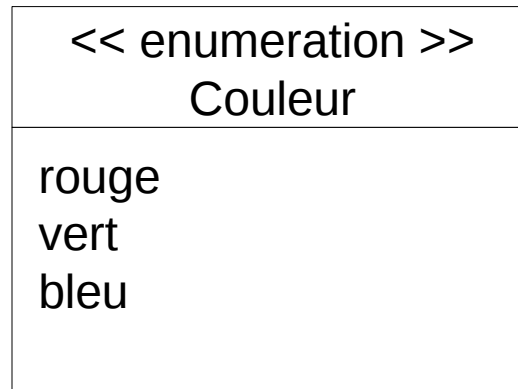
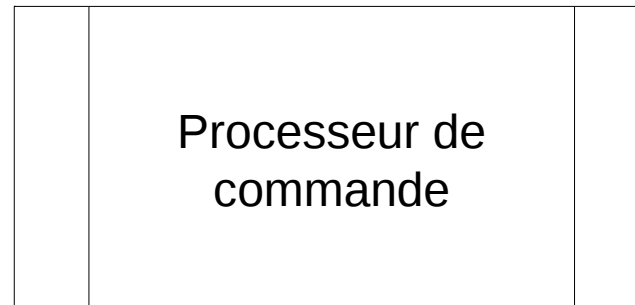


Diagramme de classe


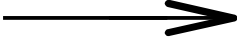
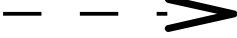




Classe active :



Instance autonome gérant ses propres threads et contrôles
Ses méthodes sont activées dans un thread ... : ex → shell

Diagramme de classe

Résumé des liens :

	Association : navigabilité (bidirectionnelle)
	Association : navigabilité unidirectionnelle
	Dépendance / Association ponctuelle
	Composition
	Agrégation
	Généralisation (Héritage)
	Implémentation