

LE LANGAGE C++

MASTER 1

À PROPOS DU PARADIGME OBJET

Jean-Baptiste.Yunes@u-paris.fr

U.F.R. d'Informatique

Université de Paris

06-2021

- **La méthode fonctionnelle** ou structurée
 - calquée sur le modèle des machines
 - des données
 - des fonctions
 - C, Pascal, Assembleur
- à ne pas confondre avec la programmation fonctionnelle

- **La méthode objet**
 - fonctionne par analogie avec le modèle du problème
 - mettre en correspondance l'espace des problèmes et l'espace des solutions

- Analyse Orientée Objets :
 - méthode d'analyse qui examine les besoins sous la perspective des objets trouvés dans le vocabulaire du domaine du problème.

Grady Booch.

Idée clé : analogie entre le modèle logiciel et le modèle physique.

Martin Abadi & Luca Cardelli.

- Ex : MACAO, OMT, Booch, OOSE

- Conception Orientée Objets :

- méthode de conception incorporant le processus de décomposition orienté objets et une notation permettant de dépeindre à la fois des modèles logiques et physiques aussi bien que statiques et dynamiques du système à étudier.

Grady Booch.

- méthode qui conduit à une architecture logicielle fondée sur les objets et leurs relations dans les sous-systèmes plutôt que sur les fonctions réalisées

Bertrand Meyer

Idée clé : souplesse/résistance/solidité du modèle.

Martin Abadi & Luca Cardelli.

- Ex : notation UML

- Programmation Orientée Objets :
 - méthode d'implémentation dans laquelle des programmes sont organisés comme des ensembles coopérants d'objets, chacun représentant une instance d'une certaine classe, et toutes les classes sont des membres d'une hiérarchie de classes unifiée par des relations d'héritage.

Grady Booch.

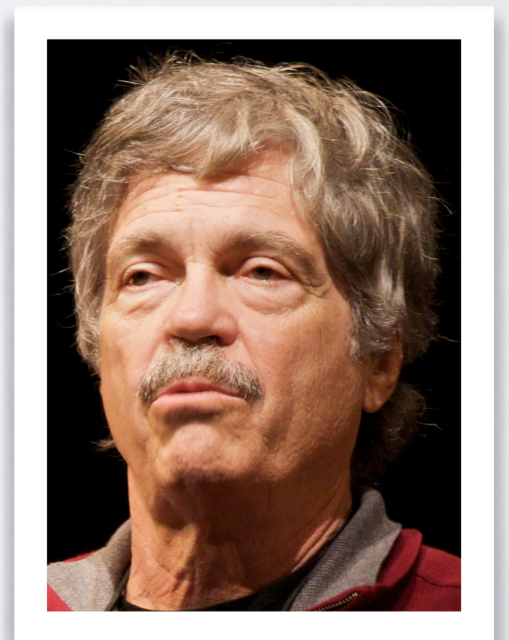
Idée clé : réutilisabilité des composants

Martin Abadi & Luca Cardelli.

- Ex : Java, C++...

« OOP to me means only messaging,
local retention and protection and hiding
of state-process, and extreme late-binding
of things. »

Alan Kay



The forgotten history of OOP

<https://medium.com/javascript-scene/the-forgotten-history-of-oop-88d71b9b2d9f>

- Intérêt ?
 - stabilité de la modélisation
 - construction itérative (couplage faible entre composants)
 - réutilisabilité
 - simplicité (relative) du modèle

UN DÉTOUR PAR LA PHILOSOPHIE

« il n'est pas question d'envisager écrire un quelconque
programme en étant dépourvu de raison »

- La philosophie s'intéresse depuis toujours à notre rapport au monde
 - lequel est constitué de ce que l'on nomme « objets »
 - de quoi s'agit-il exactement ?
- nous ne répondrons pas à cette question, mais la lecture de textes philosophiques peut nous éclairer sur la démarche constitutive de la programmation objet
 - si nous connaissons mieux les objets du monde, peut-être cela nous facilitera-t-il la fabrication de leurs avatars numériques ?

Toute société existe en créant des significations imaginaires sociales - soit, de l'imperçu immanent. Tel, le Dieu hébraïque, chrétien ou islamique; ou la marchandise. Personne n'a jamais vu une marchandise : on voit une voiture, un kilo de bananes, un mètre de tissu. C'est la signification imaginaire sociale marchandise qui fait fonctionner ces **objets** comme ils fonctionnent dans une société marchande.

Cornelius Castoriadis

Anthropologie, philosophie, politique

Conférence à l'université de Lausanne, 11/05/1989, dans Actes des colloques du groupe d'études pratiques sociales et théories, université de Lausanne, 1990.

Reproduit dans *La montée de l'insignifiance - Les carrefours du labyrinthe 4*, Éditions du seuil, 1996.

L'ontologie distingue dans les êtres dont nous avons l'expérience deux principes métaphysiques : l'essence et l'existence.

Par essence on entend ce qu'un être est : ceci, c'est du papier; je suis un homme, je possède l'essence humaine. Mais, je n'exprime pas par là tout ce qu'est cette feuille de papier, tout ce que je suis. Du donné réel je ne retiens que les caractères qu'il possède en commun avec tous les êtres de même espèce : ces caractères constituent l'essence universelle. [...]

L'essence n'implique pas qu'il existe des êtres en qui elle soit réalisée. [...] Les chimistes conçoivent des corps dont, sans en avoir vu aucun spécimen, ils connaissent parfaitement la composition, les propriétés et l'utilisation possible; il leur manque seulement de découvrir le procédé permettant de faire la synthèse de leurs éléments constitutifs. [...]

L'être de l'essence est d'être possible.

Cette possibilité devient réalité grâce à l'existence : l'existence est donc ce qui actualise l'essence. [...]

On est ainsi amené à concevoir une essence unique pouvant se multiplier dans un nombre indéfini d'individus appelés à l'existence.

Le monde sensible est le seul que connaisse le vulgaire. Il est constitué par l'ensemble des réalités perçues par les sens. On y observe des individus multiples reproduisant le même type, des violettes, des chevaux, des hommes.... et présentant des caractères communs : l'égalité, la beauté, la justice...

Au contraire, ne pouvant qu'être conçu par l'esprit, le monde intelligible est ignoré de la masse. C'est là qu'on peut contempler les types même que reproduisent les êtres du monde sensible : la Violette, le Cheval, l'Homme; ou plutôt - car ce n'est guère qu'à cela que songe Platon - l'Être lui-même dans sa pureté ou du moins ses formes les plus générales : la Bonté, la Beauté, la Justice, etc. Il les appelle des Idées, non pas qu'il les considère comme de simples représentations, mais parce qu'il voit en elles les modèles de toutes choses.

Paul Foulquié.

Que sais-je ? n°253, dixième édition, L'existentialisme, Presses Universitaires de France, 1958.

Heidegger distingue deux modes d'être des choses : les choses simplement subsistantes (Vorhanden), objets de contemplation esthétique ou théorique, et les choses telles qu'elles se donnent dans nos activités et préoccupations quotidiennes, les choses « sous la main » (Zuhanden). De prime abord et le plus souvent, nous entretenons un rapport pratique avec elles [...] Elles m'apparaissent d'emblée avec le « pour quoi » de leur utilité [...] Or, ce régime d'utilité a tendance à conquérir, au-delà des outils, tout le monde environnant [...] Pour que les choses puissent apparaître pour ce qu'elles sont, précise Heidegger, il faut une rupture dans l'ordre de la préoccupation : qu'un outil soit inutilisable [...] C'est donc en rompant avec le mode de la disponibilité, soutient Heidegger, que nous pouvons appréhender les choses pour elles-mêmes, que nous pouvons les voir surgir dans leur simple subsistance et adopter face à elles un regard contemplatif.

Martin Legros

Dossier *Apprendre à l'ère d'Internet*, Philosophie magazine, n° 62, 09/2012.

- Ludwig Wittgenstein dans *Tractatus logico-philosophicus* (1922), se pose la question du sens.
 - se faisant, il formalise logiquement ce qui fait sens
 - il propose de décomposer le monde en faits;
« Magic Johnson est grand »
 - les faits se décomposent en objets
 - les particuliers « Le bureau »
 - les propriétés « Robert est lourd »
 - les relations « Nathalie adore la soupe »

- Ludwig Wittgenstein :
 - 2.02 **L'objet** est simple.
 - 2.0233 Deux **objets** de même forme logique - leurs propriétés externes mises à part - ne se différencient l'un de l'autre que parce qu'ils sont distincts
 - 2.0271 **L'objet** est le fixe, le subsistant; la configuration est le changeant, l'instable.
 - 2.061 Les états de **choses** sont mutuellement indépendants.
 - 3.22 Le nom est dans la proposition le représentant de **l'objet**.

A map of the Western Atlantic Ocean showing the Bermuda Triangle. Three red lines connect the vertices of the triangle: Miami, Florida; Bermuda; and Puerto Rico. The map includes labels for 'îles des Bermudes' (Bermuda), 'mer des Sargasses (océan Atlantique)' (Sargasso Sea (Atlantic Ocean)), 'Floride' (Florida), and 'Porto Rico'.

LE TRIANGLE SÉMIOTIQUE

Sémiotique : branche de la philosophie du langage

la pensée



Sens (signifié, notion,
abstrait, idée, concept,
pensée)

Se rapporte

Symbolise

la réalité



le langage

Une maison

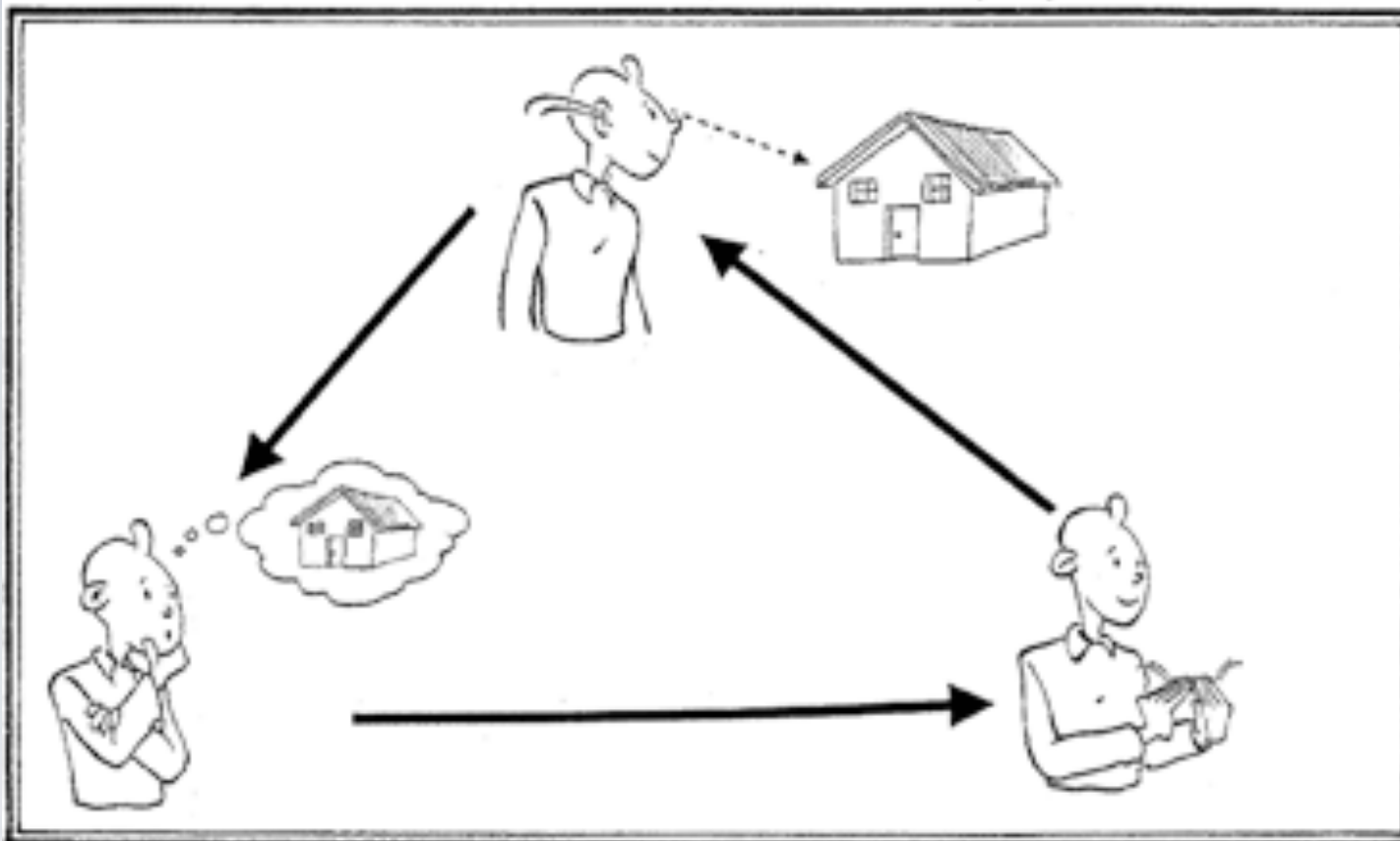
Objet (chose, concret, réalité
(domaine extralinguistique))

Représente

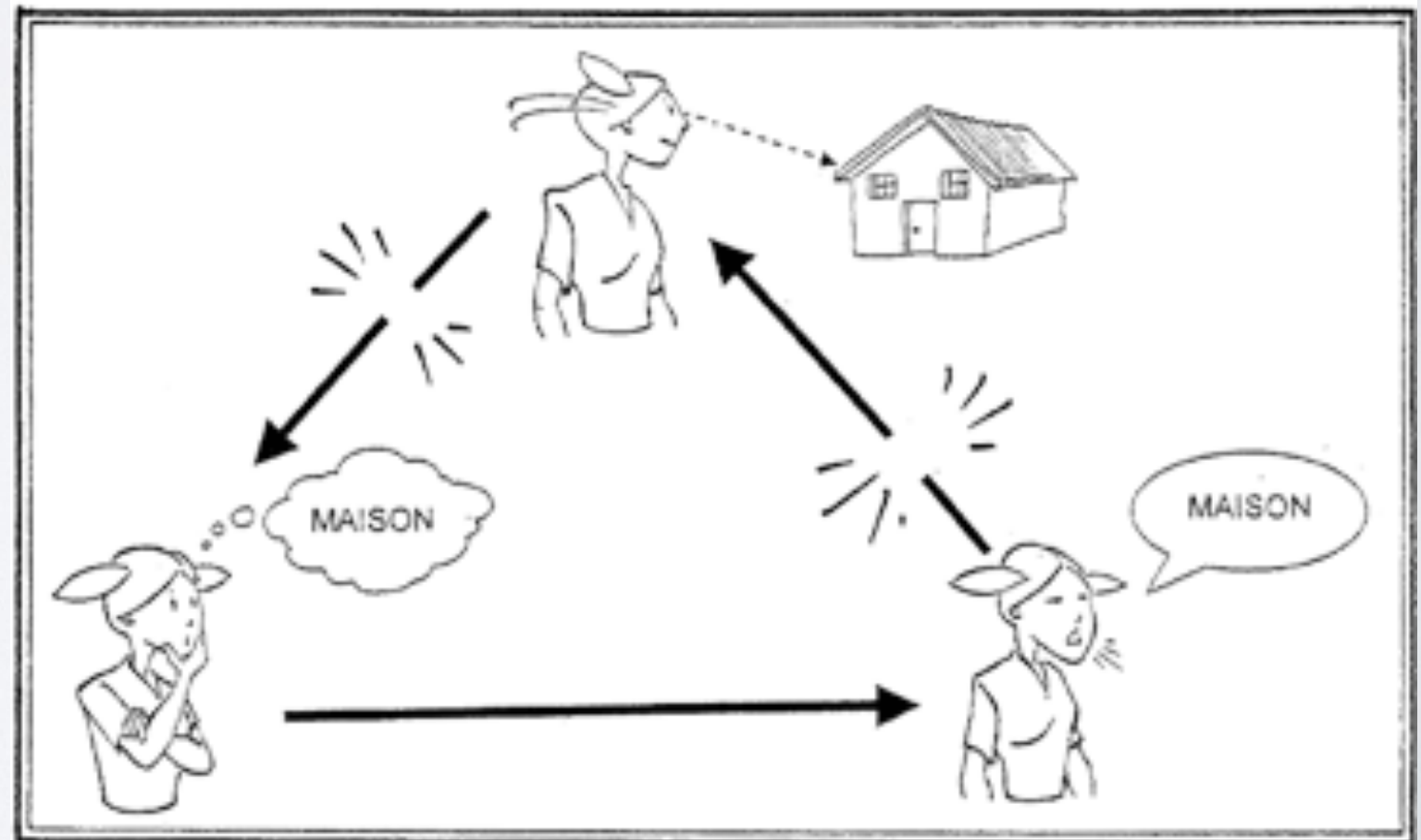
Nom (mot, signifiant,
symbole, langage)

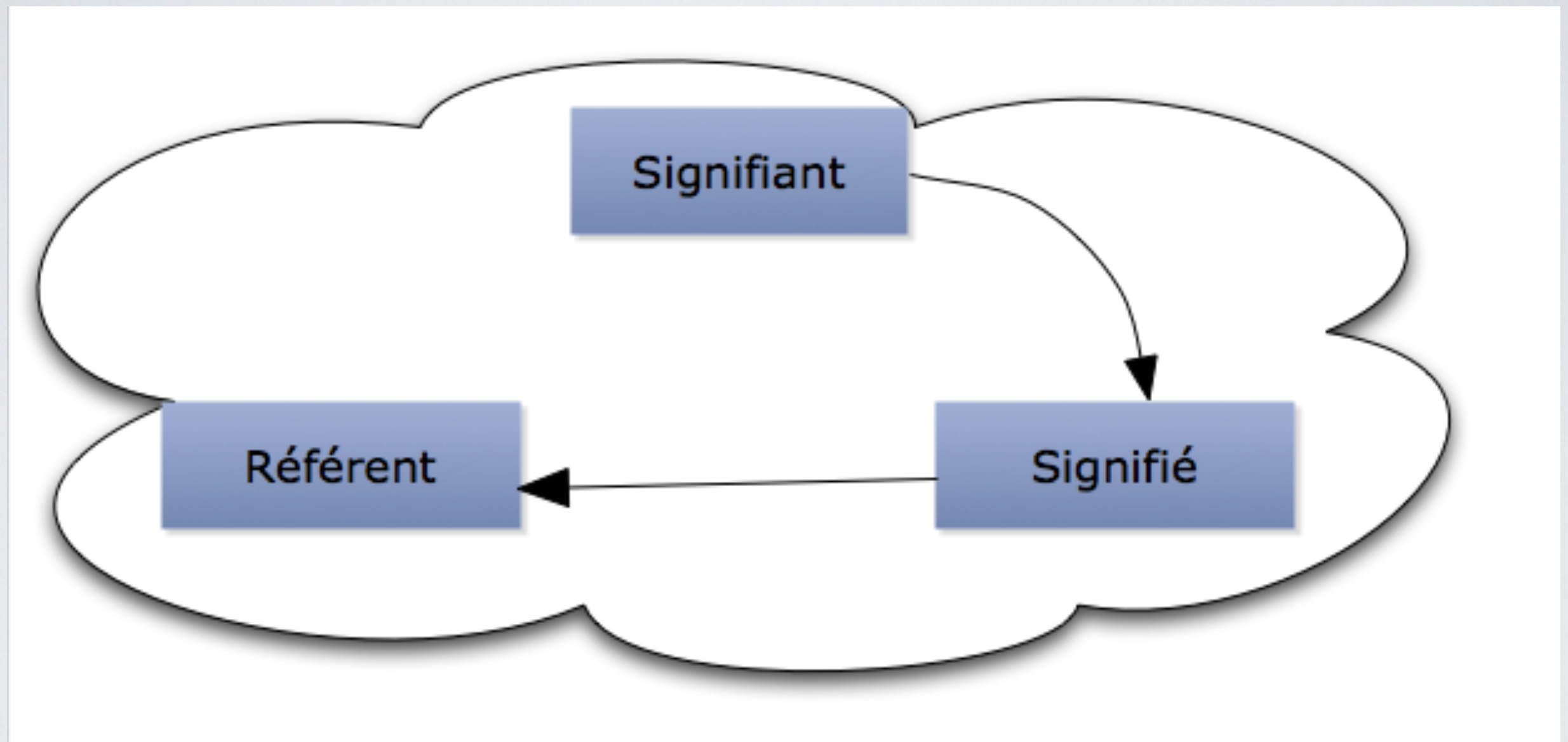
Le triangle sémiotique

EN LANGUE DES SIGNES (LS)



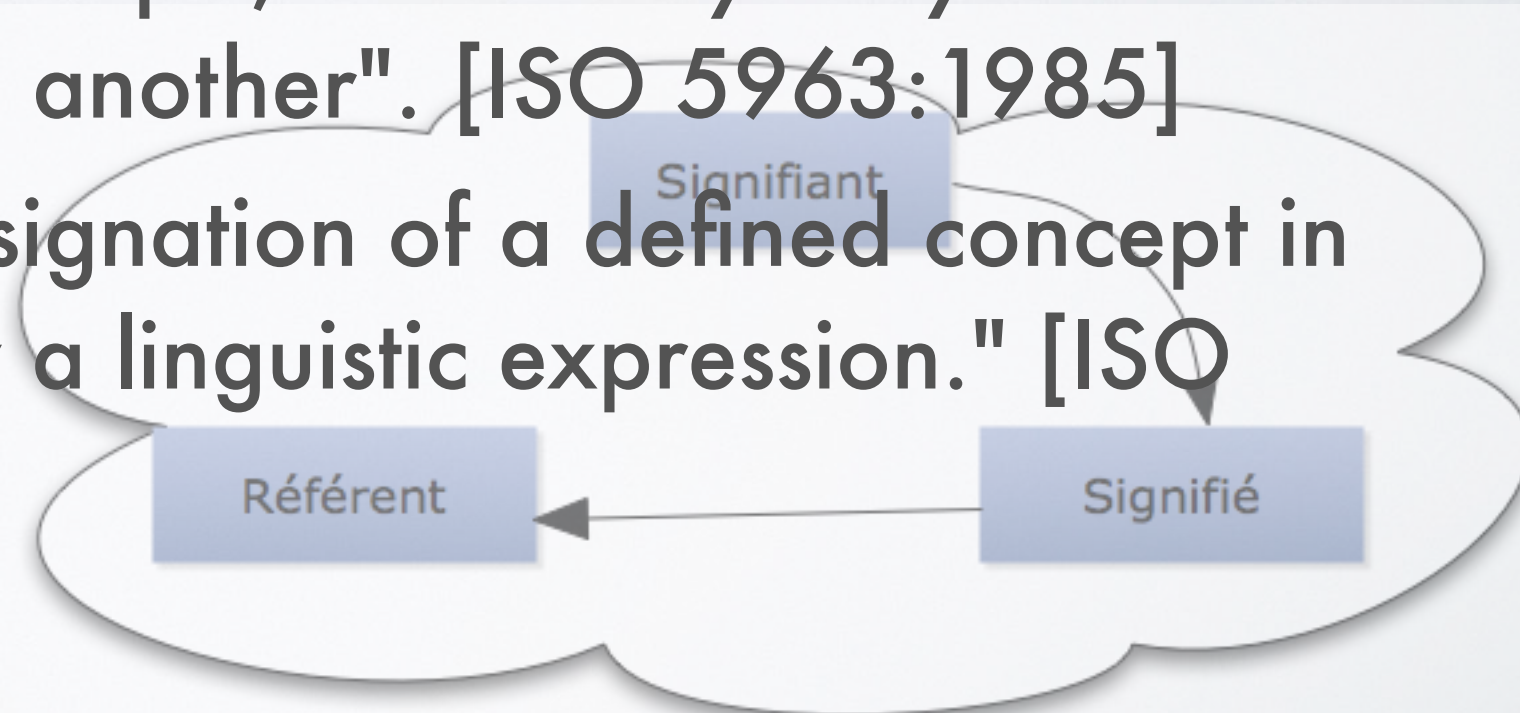
EN LANGUE VOCALE (LV)



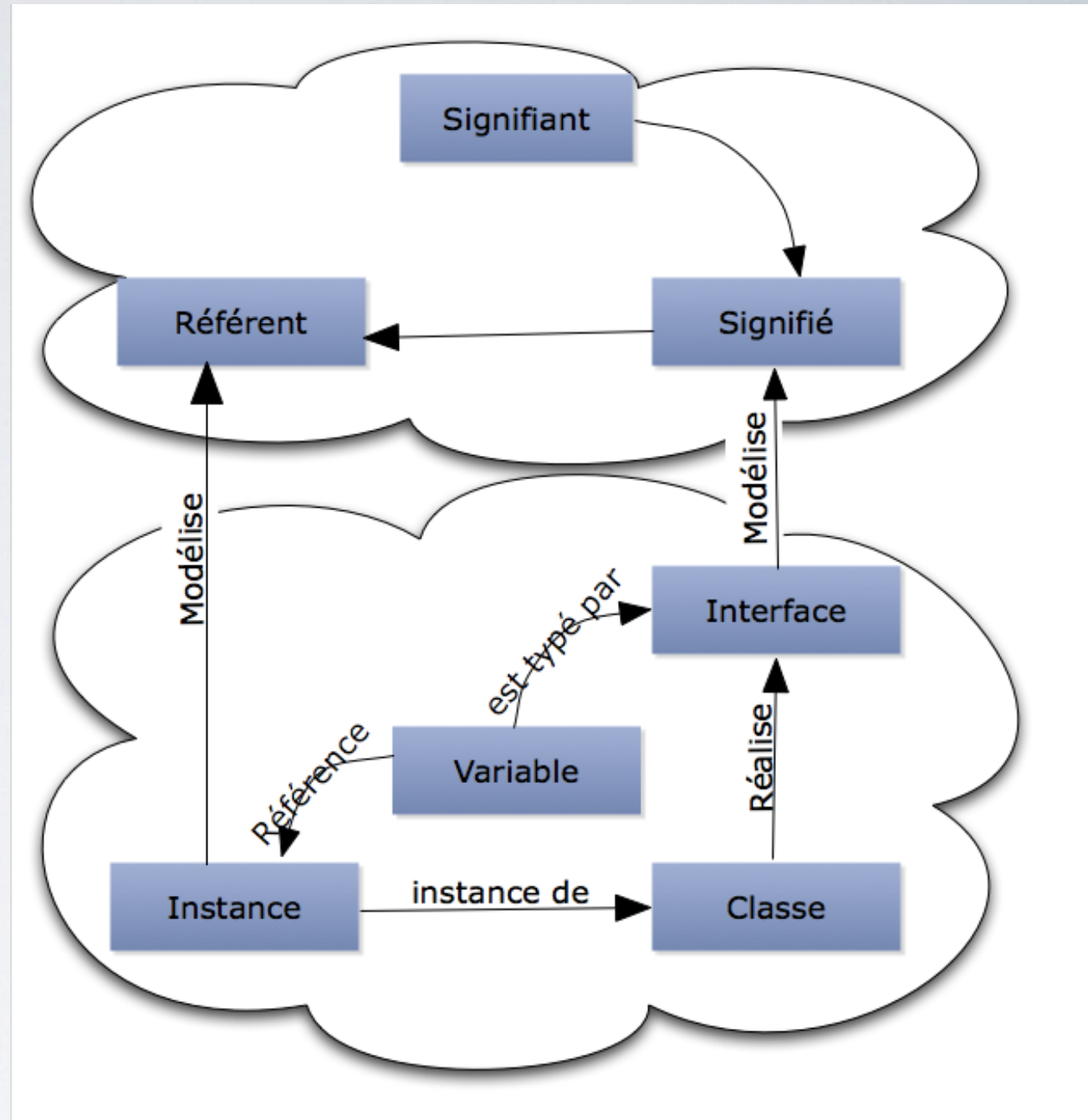


Pas de relation directe entre le mot et l'objet. Les mots, pour désigner les objets, passent par la médiation du concept...

- **Objet/référent** : "any part of the perceivable or conceivable world." [ISO 1087]. Référents matériels (mon marteau, ma voiture, la Halle aux Farines). Référents immatériels (le temps, la nation, la liberté...)
- **Concept/signifié** : "A unit of thought constituted through abstraction on the basis of properties common to a set of objects. [ISO 1087] The semantic content of a concept can be re-expressed by a combination of other and different concepts, which may vary from one language or culture to another". [ISO 5963:1985]
- **Terme/signifiant** : "Designation of a defined concept in a special language by a linguistic expression." [ISO 1087]



- Conception :
- Interface
- Réalisation :
- Programmation



- Du triangle à la conception, puis à la programmation

QUELQUES ÉLÉMENTS PARADIGMATIQUES

Un paradigme est une représentation du monde, une façon de voir les choses. En programmation, cela fait référence à un style fondamental de penser les constituants des programmes.

- Éléments majeurs
constitutifs du modèle à
objets :

- l'abstraction
- l'encapsulation
- la modularité
- la hiérarchie

Grady Booch

- Éléments mineurs :
 - le typage
 - la simultanéité
 - la persistance

- Éléments essentiels
constitutifs du modèle à
objets :

- L'identité
- la classification
- le polymorphisme
- l'héritage

James Rumbaugh

- Éléments annexes :
 - l'abstraction
 - l'encapsulation

- **L'abstraction :**
 - fait ressortir **les caractéristiques essentielles d'un objet** (qui le distinguent de tous les autres genres d'objets) et donc procure des frontières conceptuelles rigoureusement définies, relativement au point de vue de l'observateur.

En algorithmique on travaille sur des types abstraits, i.e. on parle de liste d'éléments sans préciser grand chose de la réalité de ces choses.

Autre exemple, lors du passage du permis de conduire on vérifie que le candidat sait conduire une voiture, mais on ne dit jamais laquelle; c'est heureux car cela lui permet par la suite de conduire n'importe quelle voiture y compris celles qui n'existaient pas encore lors de l'examen.

- **L'encapsulation** - Information hiding :
 - procédé qui cache l'ensemble des détails d'un objet qui ne font pas partie de ses caractéristiques essentielles.

C'est un concept très courant et utile. Dans la vie réelle on parle de boîte noire (un appareil électro-ménager est une boîte noire, i.e. on n'en voit pas les détails de fonctionnement), un composant électronique est une boîte noire.

- **La modularité :**
 - propriété d'un système qui a été décomposé en un ensemble de modules cohérents et faiblement couplés.

Dans l'industrie, ceci permet de nettement différencier les tâches à l'étape de réalisation et de confier celle-ci à des équipes ou des sous-traitants indépendants.

- **La hiérarchie :**
 - rangement ou ordonnancement des abstractions.
La hiérarchie est essentielle car elle permet de raisonner à différents niveaux conceptuels; opération courante dans la vie de tous les jours et qui fait la richesse du langage humain; le paradigme objet permet d'en obtenir des bénéfices équivalents à l'intérieur des programmes.

Par exemple, on est pas au même niveau de description et de détail lorsque qu'on parle de :

Néofélidés Félidés actuels	Panthérinés	Lion, Tigre, Panthère, Jaguar. Once.
	Félinés	<u>Chat</u> , Ocelot, Serval, Jaguarondi, Margay. Lynx, Caracal. Puma.
	Acinonychinés	Guépard.
Paléofélidés Groupes disparus	Prolailurinés	Prolairus.
	Machairodontinés	<u>Smilodontinés</u> : Smilodon, Méganthéréon.
		<u>Homotheriinés</u> : Homotherium, Machairodus. <u>Metailurinés</u> : Dinofelis, Metailurus.

- Le **typage** :
 - est le fait d'imposer la classe d'un objet, de sorte que des objets de types différents ne puissent être intervertis, ou alors de façon très restrictive.
- La **simultanéité** :
 - distingue les objets actifs des inactifs.
- La **persistance** :
 - propriété des objets à transcender le temps ou l'espace.

- **L'identité :**
 - les objets sont des entités discrètes distinctes.
- **La classification (à la Rumbaugh) :**
 - les objets de même structure (attributs et opération) sont regroupés en classes.
- **Le polymorphisme :**
 - une même opération peut se comporter différemment selon l'objet sur laquelle elle est appliquée

Tous les animaux qui crient ne le font pas de la même manière donc faire crier un animal ne produit pas le même résultat sur tous

- **L'héritage :**
- **partage d'attributs et d'opérations
basé sur une relation hiérarchique**

La classification hiérarchique étant généralement obtenue par factorisation d'attributs ou d'opérations communes, il s'agit donc d'un héritage structurel : si une figure géométrique est définie par la propriété de pouvoir être translatée, alors un triangle (qui est une figure géométrique) peut l'être. Idem pour un attribut de position par exemple.

- Les **objets** (d'après Pierre-Alain Muller) :
 - Un objet est une entité atomique formée de l'union d'un état et d'un comportement. Il fournit une relation d'encapsulation qui assure à la fois une cohésion interne très forte et un faible couplage avec l'extérieur. Il révèle sa vraie nature par l'envoi de messages lorsqu'il s'insère dans un scénario de communication. Un objet possède une identité.

Objet = État + Comportement

- **L'état :**
 - union des valeurs instantanées de tous les attributs d'un objet. Un attribut est une information qui qualifie l'objet qui le contient.
- **Le comportement :**
 - union des compétences d'un objet (actions et réactions d'un objet). Un atome de compétence est appelé une opération. Les opérations sont déclenchées à la réception d'un message.

- **L'identité :**

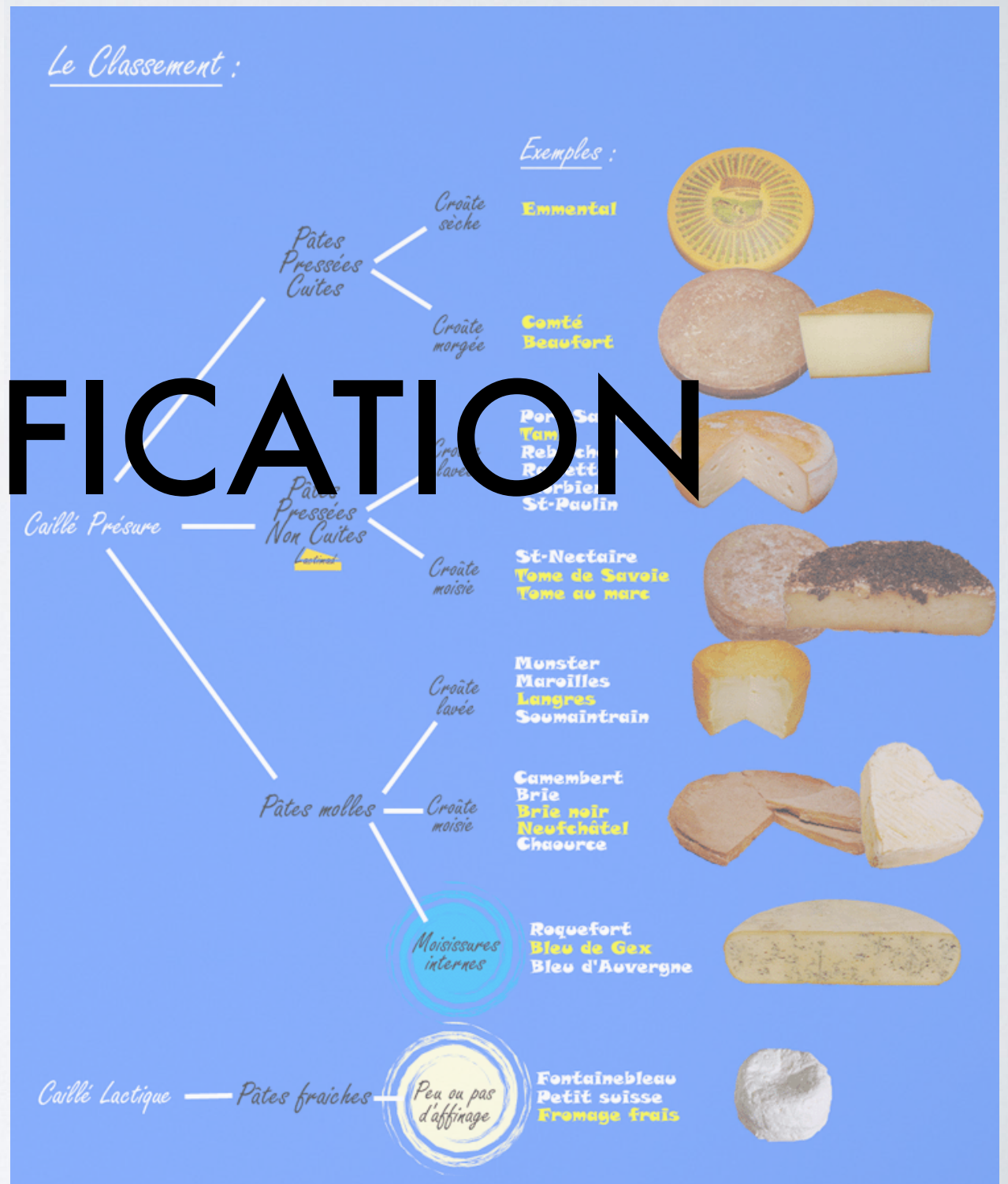
- permet de distinguer deux objets de façon non ambiguë. C'est un concept. Chaque objet possède implicitement une identité.

Dans les cas ordinaires, la machine objet confond identité et localisation. Mais dans le cas du distribué, l'identité d'un objet permet de retrouver sa localisation à l'aide d'un répertoire.

- En Java l'identité d'un objet local est son adresse en mémoire, en C++ aussi.

[illegible]

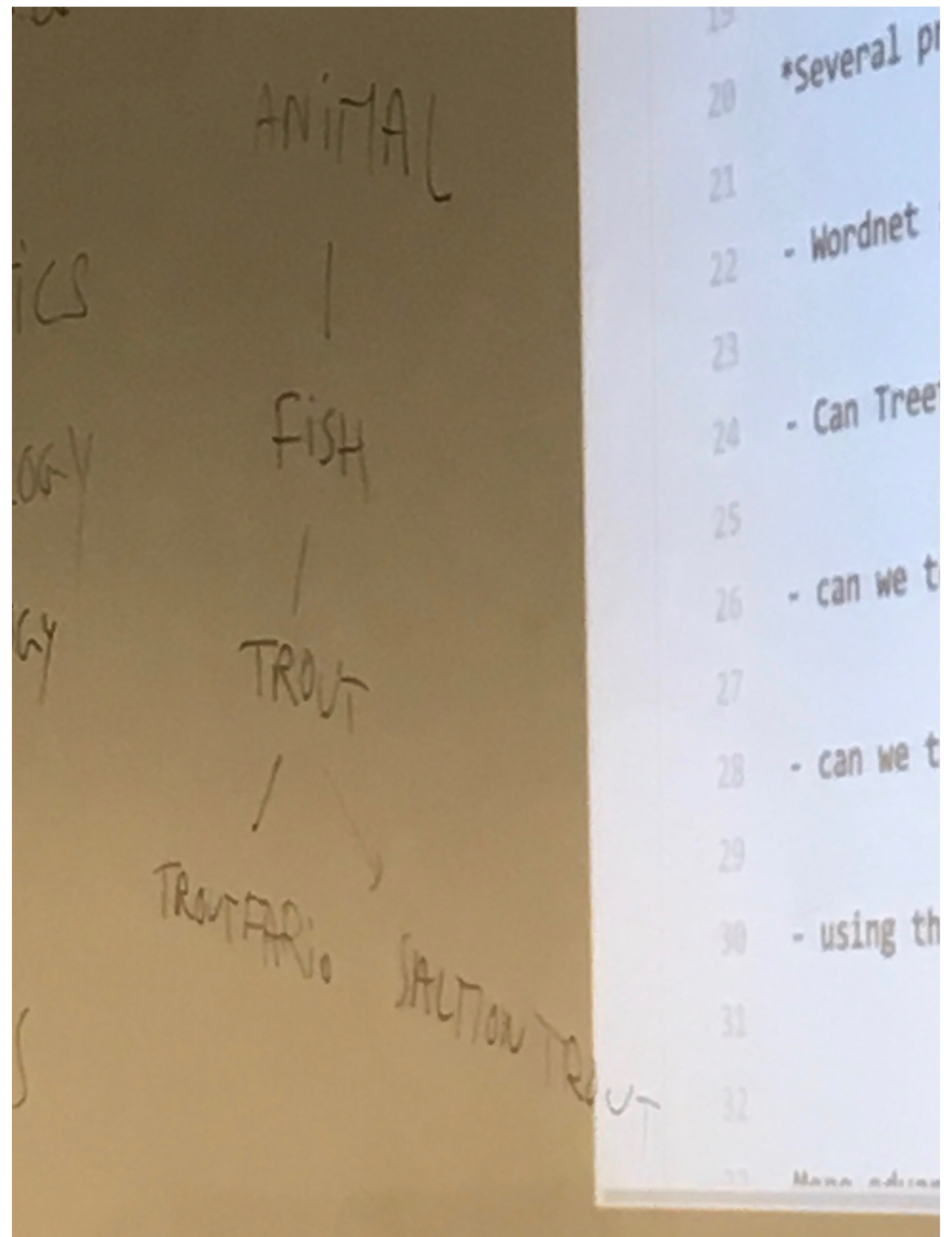
LA CLASSIFICATION



- La classification est un processus extrêmement courant
- on passe notre temps à classifier

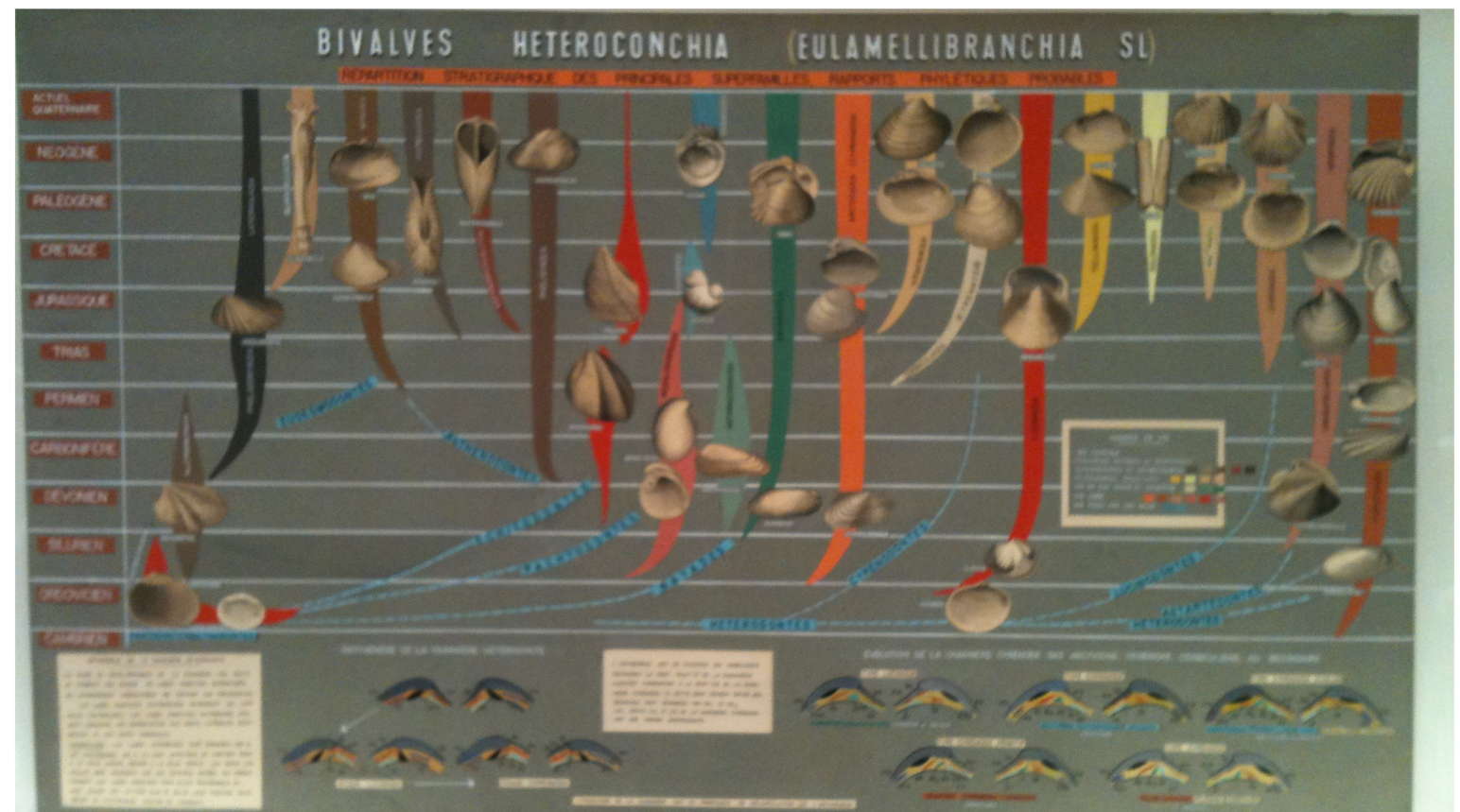
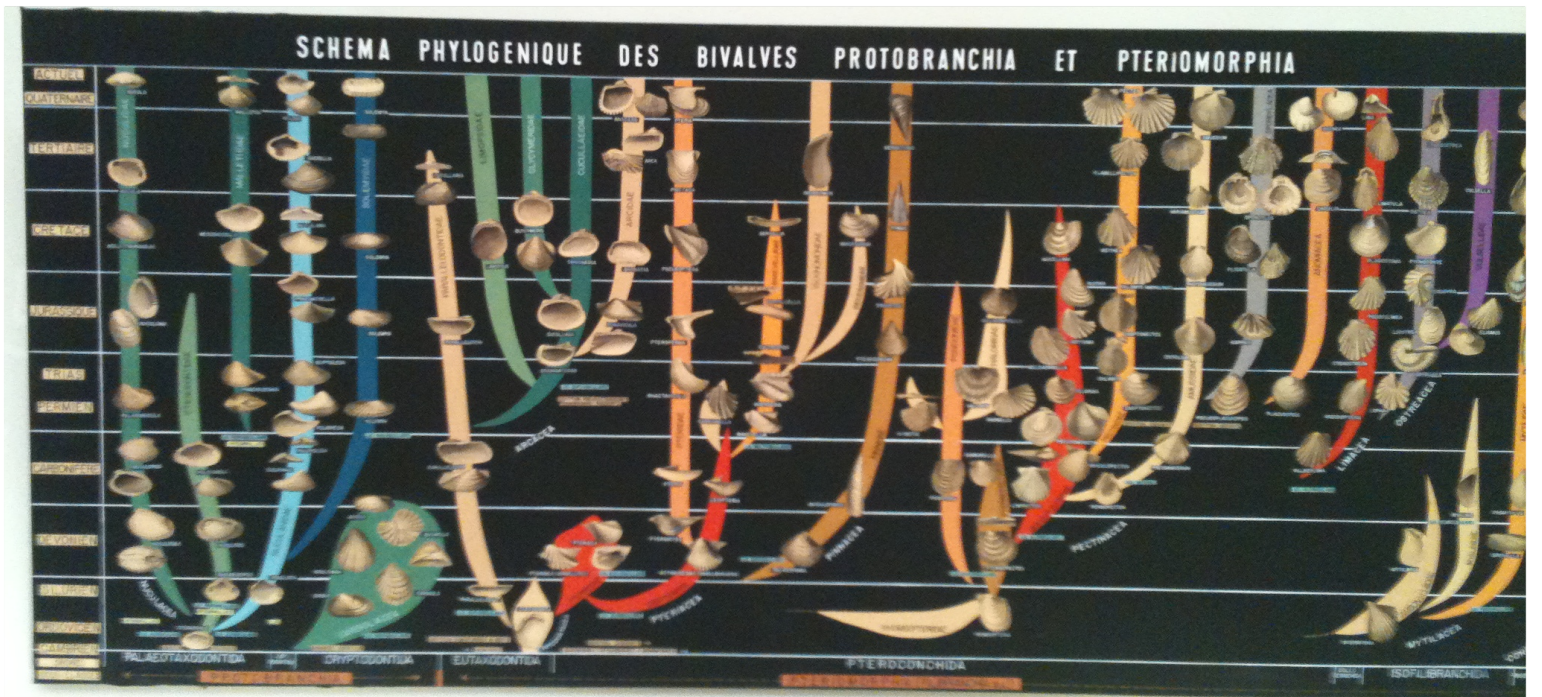
Élaboré en cours de linguistique pour parler d'ontologie sémantique

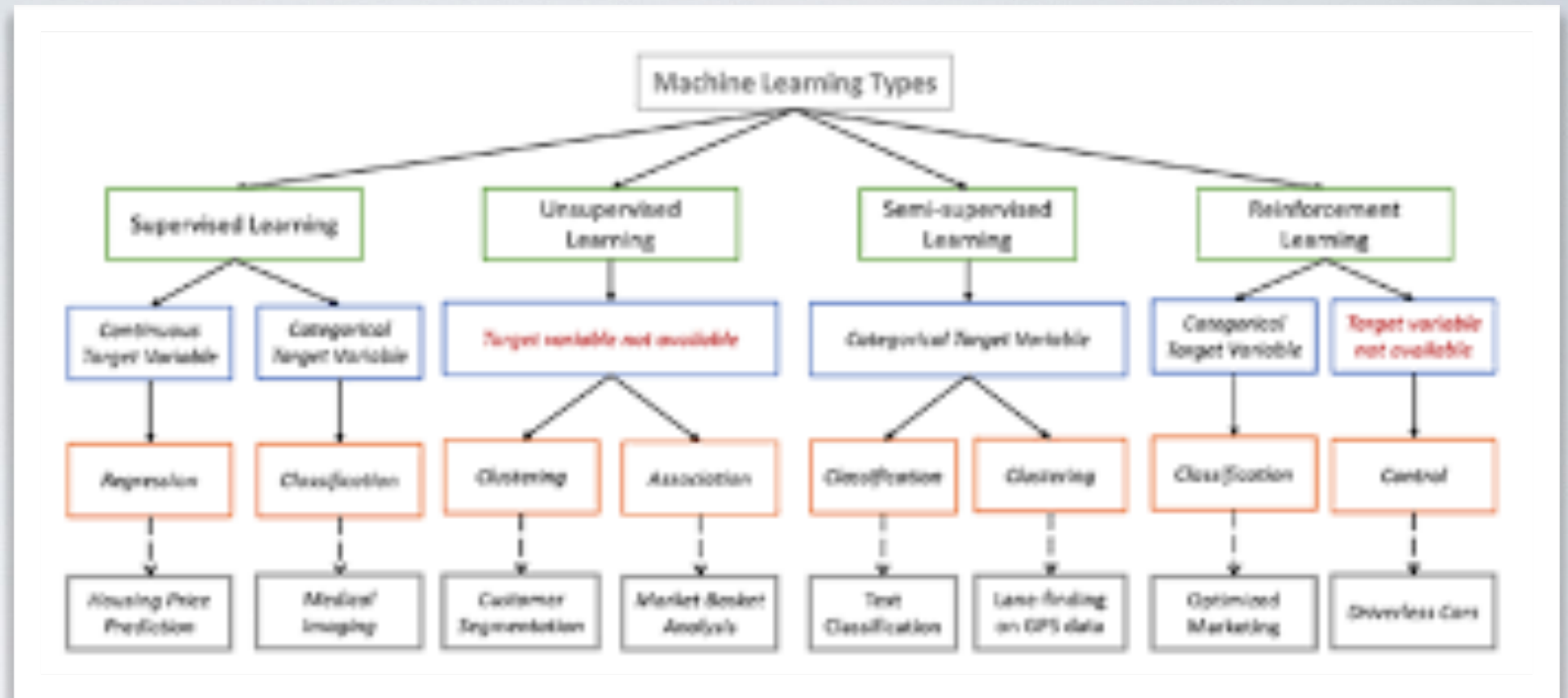
N. Ballier - 2019



Classification phylogénétique

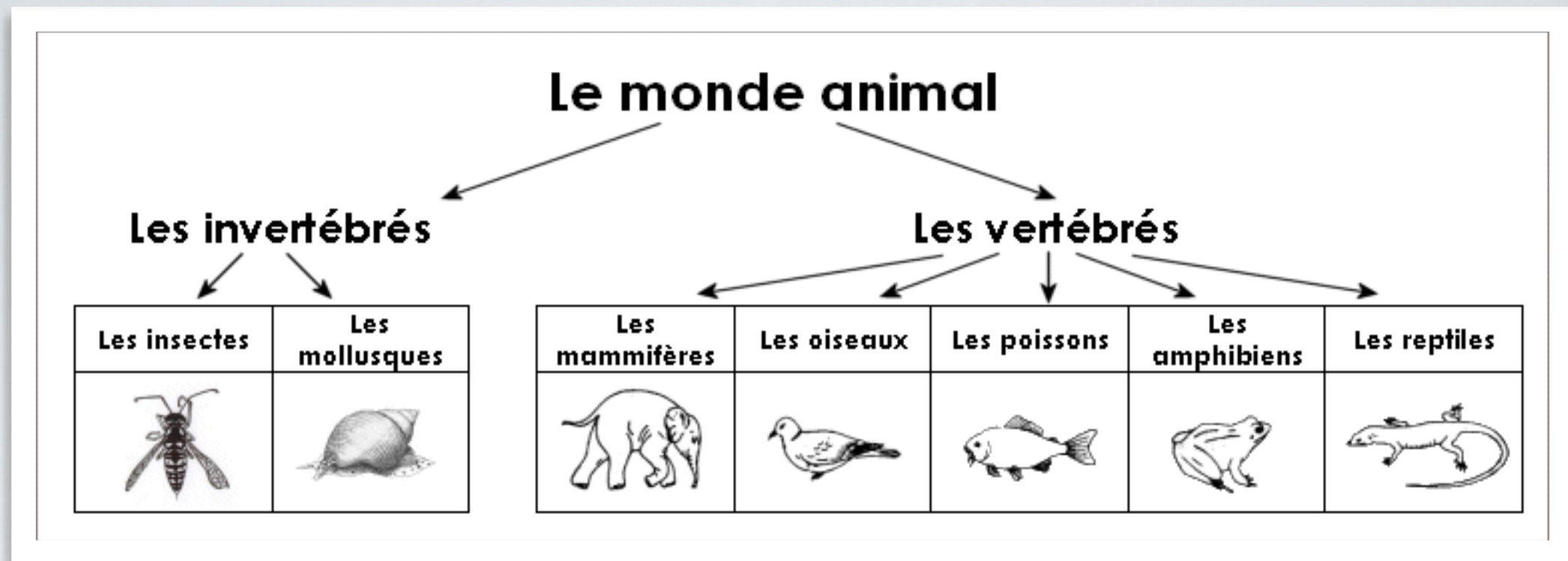
Musée de
paléontologie - 2010





Classification des
algorithmes
d'apprentissage

source inconnue



Classification classique
des espèces

Support pédagogique
pour école primaire

La classification par Guillaume Lecoindre

- C'est une opération courante, mais qu'il faut bien comprendre et maîtriser pour éviter d'aboutir à des absurdités. Erreurs très grave mais très courante :

la confusion entre généralisation (isa), typage (kindof) et composition (partof)

is a	relation entre classes inclusion d'ensembles
kind of	relation d'un objet à une classe appartenance d'un élément à un ensemble
part of	relation de composant à composé

- Quelques éléments terminologiques

Classe	Abstraction	Concept	Ensemble
Objet	Réification <i>Chosification</i>	Chose	Élément

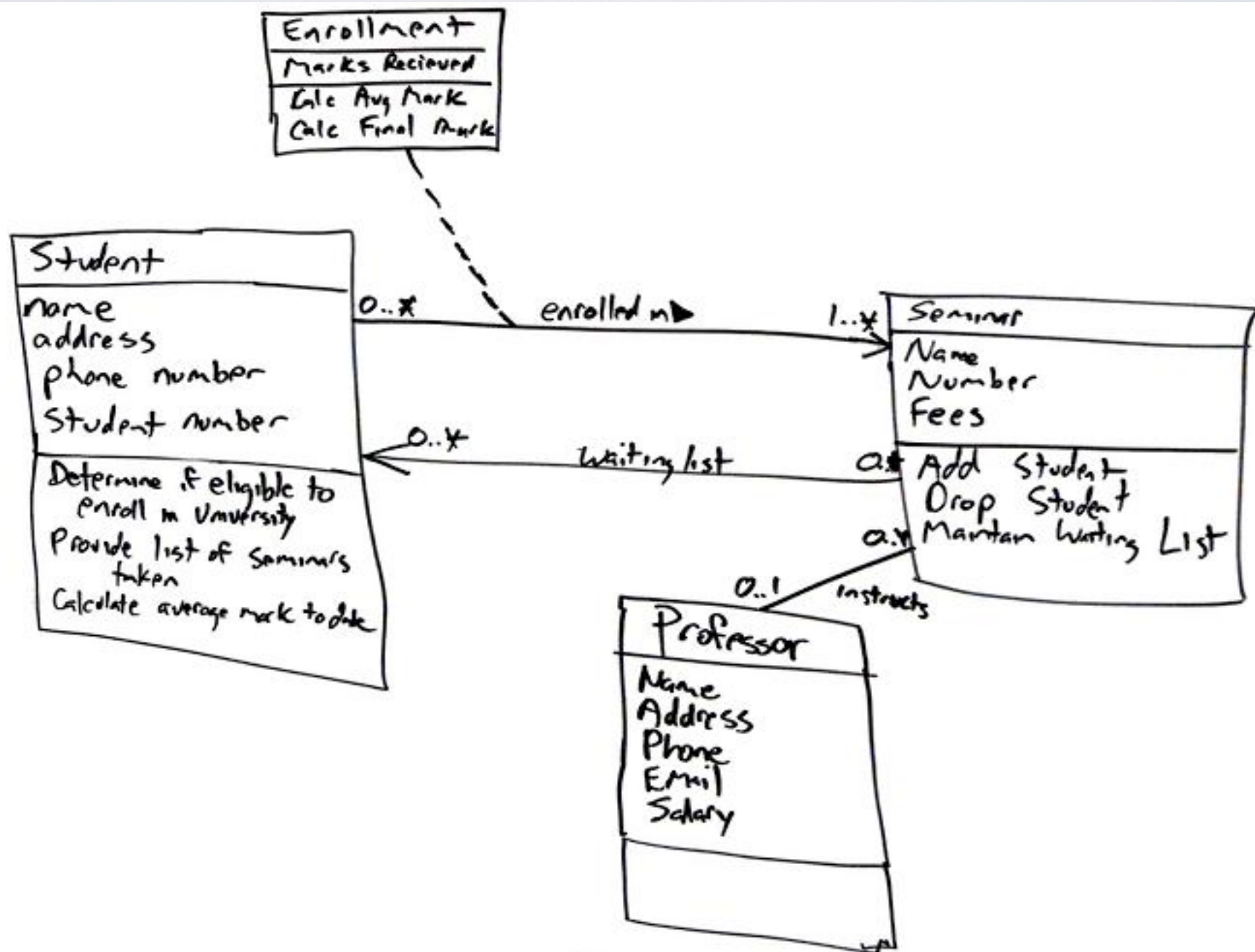
ODÉLISATION

Map of the Paris 13 district showing the University of Paris 13 buildings highlighted in red. The map includes street names like Avenue de France, Rue Marie-Andrée Lagroua Weill-Haile, and Rue Elsa Morante. It also shows public spaces like 'jardin public' and 'Watt Biopark'. The Seine river is at the bottom. A legend indicates bus lines (62, 89, 325, 64) and RER C and Métro 14 M stations.

- Le langage couramment employé pour décrire la structure d'une application dans le monde objet est UML : Unified Modeling Language
- Il s'agit bien d'un langage (graphique) permettant d'exprimer différents aspects structurant l'application dont :
 - les objets manipulés et leurs relations
 - les abstractions correspondantes : concepts et relation entre concepts
 - l'évolution des objets
 - les interactions entre objets

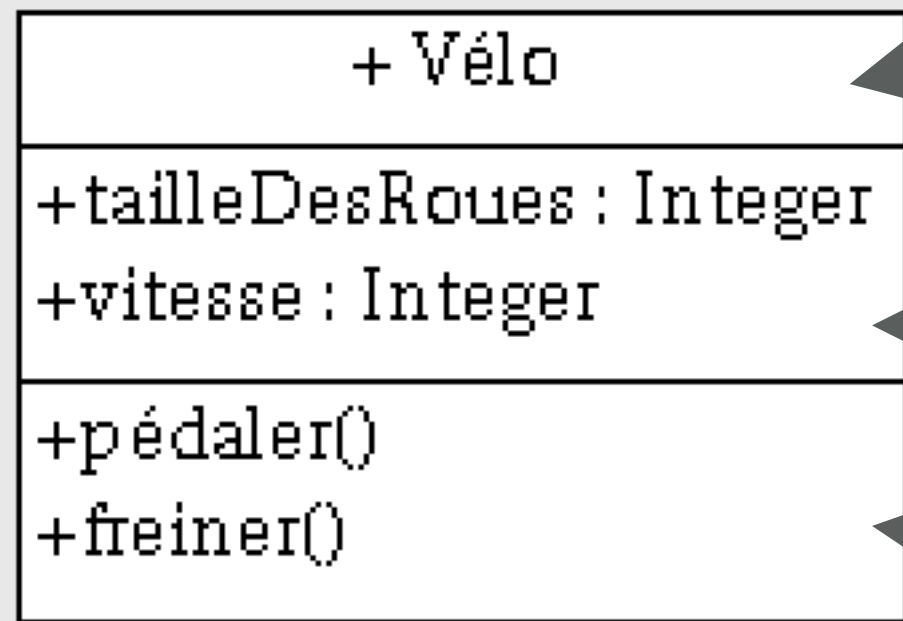
- **Le diagramme d'objets**
 - décrit un certain nombre d'objets identifiés comme pertinents dans un certain contexte
 - décrit les relations existantes entre les objets identifiés
- un diagramme d'objet ne contient pas tous les objets (ce n'est ni toujours possible, ni raisonnable)
- un diagramme d'objets modélise des faits précis et particuliers
- l'abstraction des diagrammes d'objets conduit aux diagrammes de classes

- **Le diagramme de classes**
 - décrit les concepts pertinents
 - décrit les relations entre concepts
- Un diagramme de classe ne contient pas toutes les classes (pas souhaitable, ni possible), mais toute classe doit être dans un diagramme de classe (ie. tous les concepts doivent être décrits!)
- Attention : le diagramme de classe ne « tombe » pas du ciel! Il est obtenu en faisant une liste des objets du système puis en opérant une classification de ces objets...



- L'analyse conduit à concevoir des classes UML
- c'est-à-dire une description des responsabilités, du comportement et du type d'un ensemble d'objets
- on rappelle que les objets sont caractérisés par leurs attributs et opérations

- Si l'on pense qu'un vélo est caractérisé par le diamètre de ses roues et que l'on peut pédaler et freiner pour contrôler sa vitesse, sa représentation UML peut être la suivante :



nom du concept

attributs (typés ou non)

opérations (typées ou non)

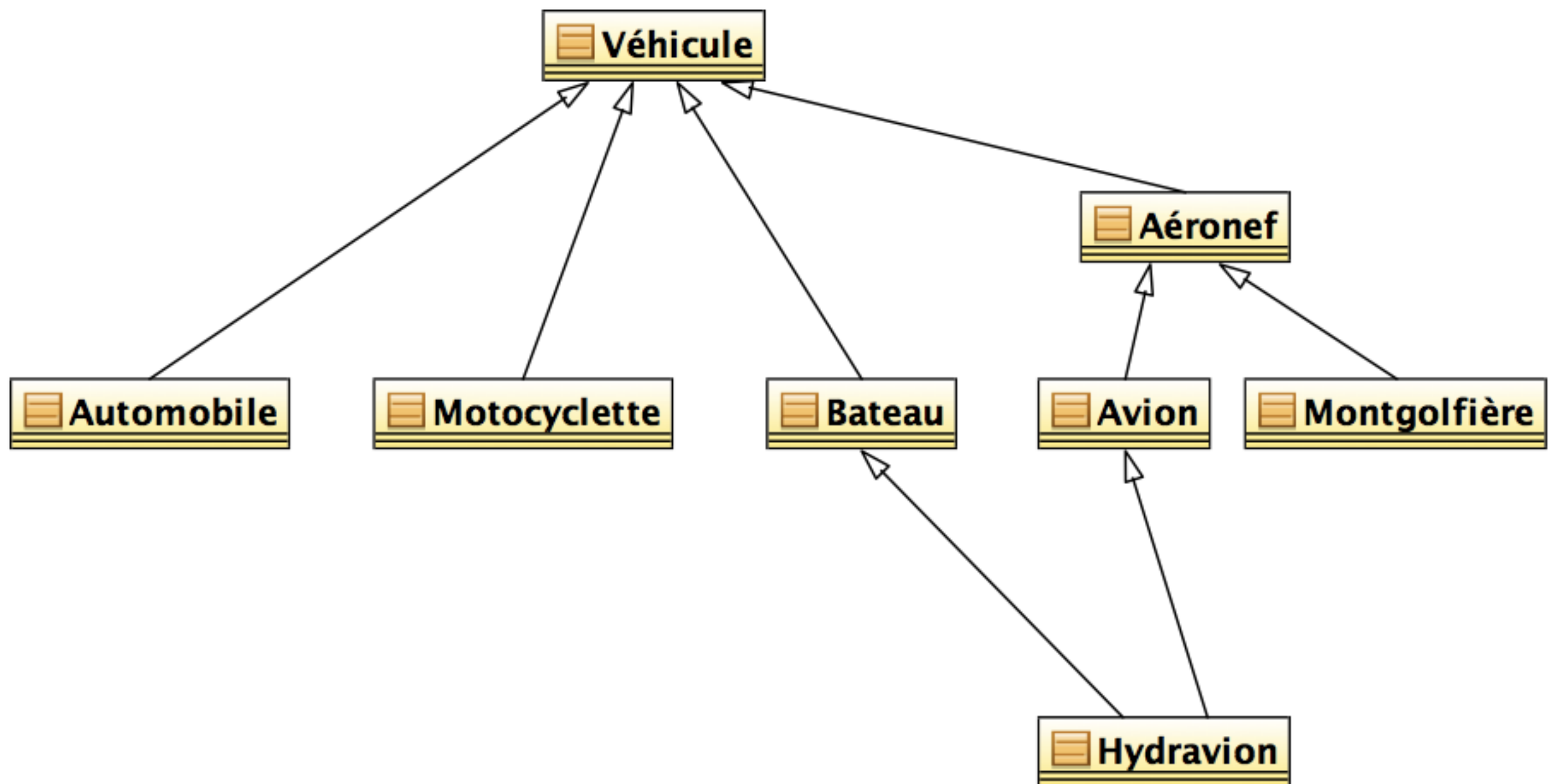
- Les relations entre classes sont de genres différents :
 - **généralisation/spécialisation**
 - induite par est-un/est-une/est-une-sort-de
 - exemple : une voiture est un véhicule
 - **agrégation/composition**
 - relation tout/partie, ensemble/élément
 - exemple : une maison est faite de pièces
 - **association**
 - simple lien
 - exemple : un frère et sa sœur

- Il est très important de ne pas confondre ces différentes relations
- Les identifier correctement est une étape essentiel du processus d'analyse

La **généralisation/spécialisation**

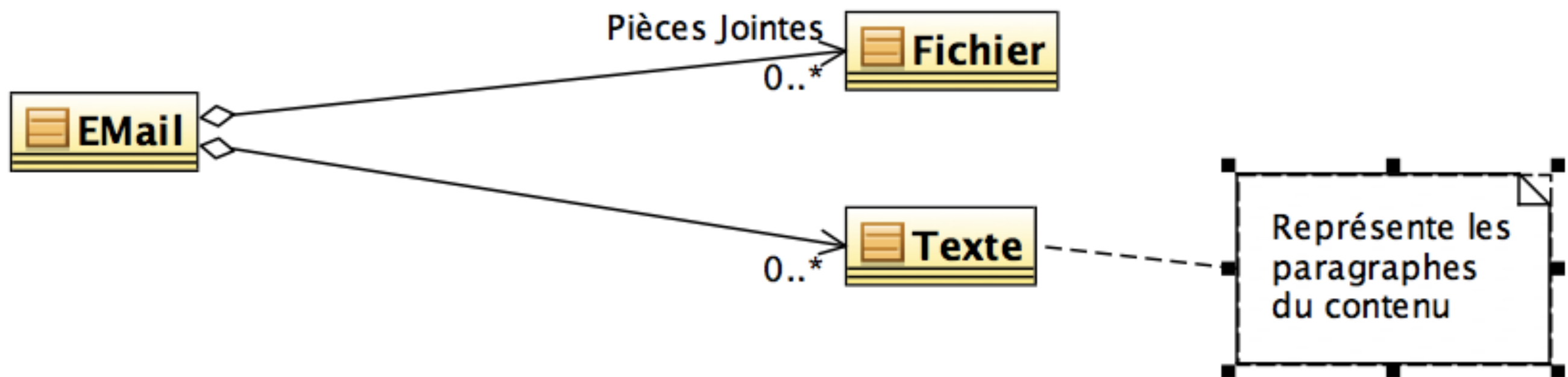
- la généralisation peut être obtenue par factorisation
- la spécialisation peut être obtenue par extension
- Cette relation autorise l'emploi du **Principe de substitution de Liskov** [1987]
 - If S is a subtype of T , objects of type S should behave as objects of type T , if they are treated as objects of type T
- Ce principe est trop contraignant, on peut lui préférer, le **Principe de substitutivité modulaire** de Narbel [2005]
 - Si $P(T)$ est un programme exprimé en les termes du type T , on peut substituer à toute instance de ce type T , une instance d'un type S sous-type de T

- tout le monde peut comprendre le sens d'un tel diagramme



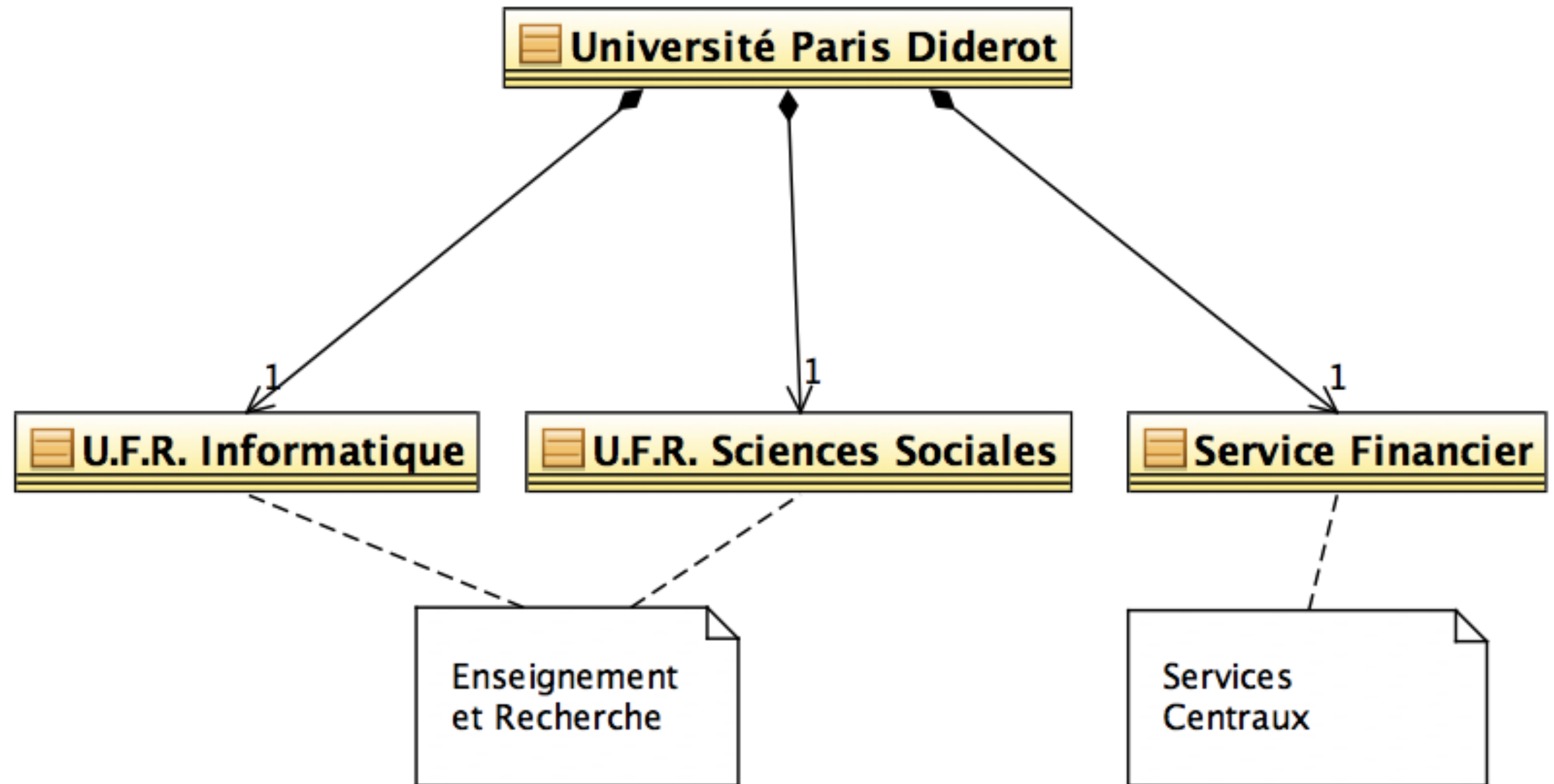
L'agrégation :

- l'agrégation est une relation de nature asymétrique reliant des ~~éléments~~ composants à un ~~ensemble~~ composé
- le couplage est fort et il y a subordination
- la vie des éléments est indépendante de celle de l'agrégat
- un élément peut appartenir à plusieurs agrégats
- ex : un e-mail peut agréger des documents joints



La composition :

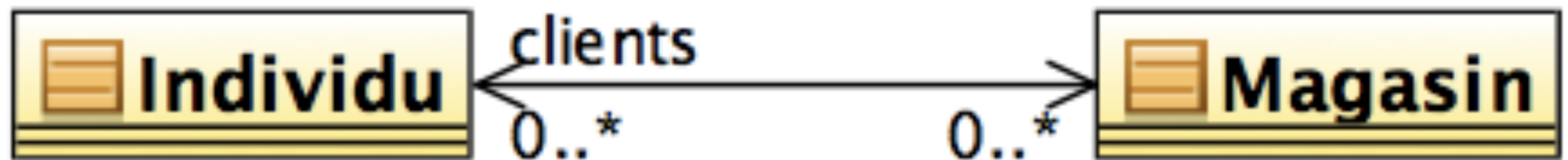
- c'est une agrégation forte (agrégat de valeurs)
- une destruction du composé entraîne la destruction des composants
- un composant ne peut appartenir qu'à un seul composé
- ex : les pièces d'une maison, l'UFR d'Informatique de Paris Diderot



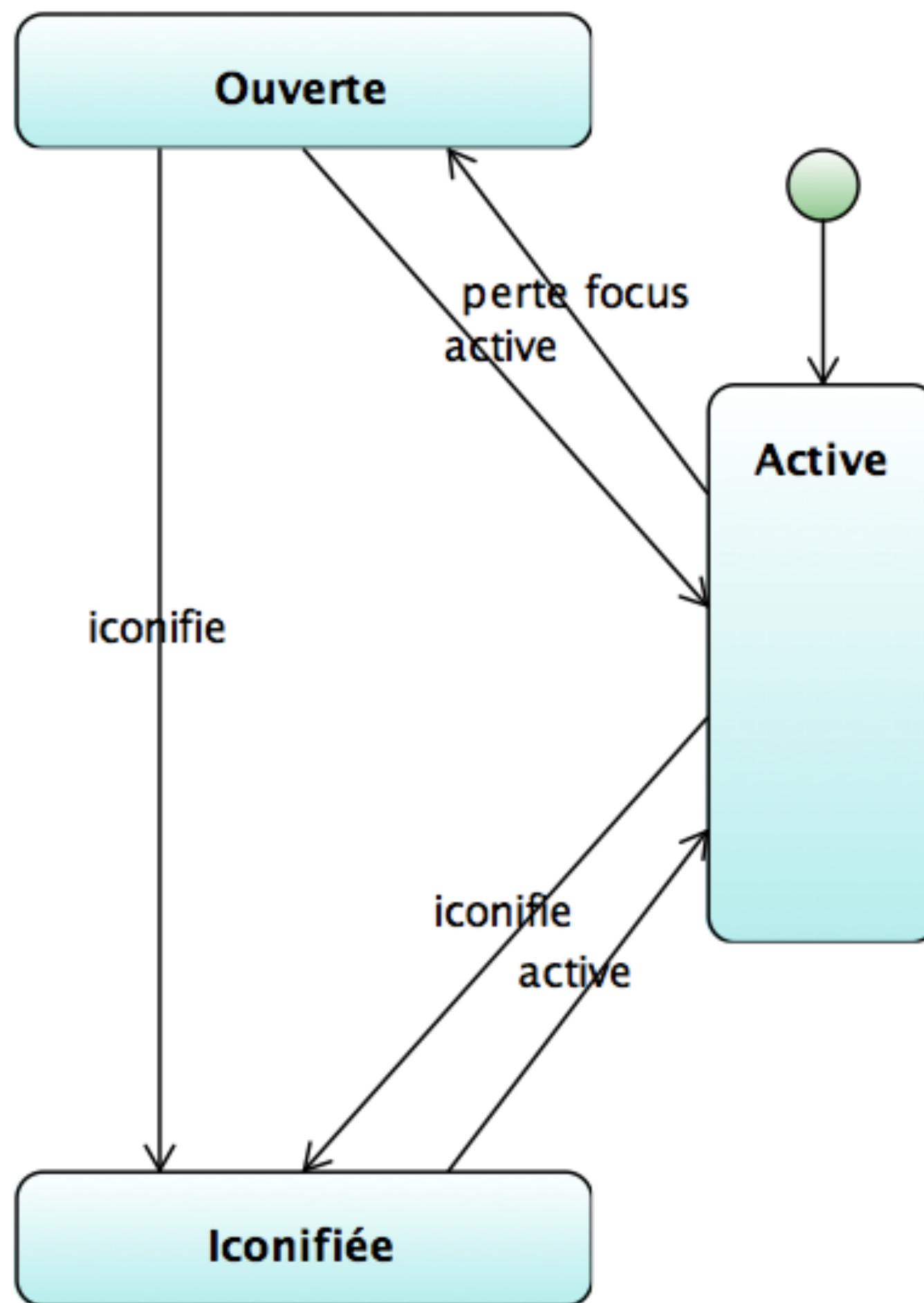
agrégat vs. composition :

- on utilise parfois la composition pour représenter un agrégat afin de simplifier la gestion du cycle de vie des composants.
- dans la réalité une molécule est un agrégat d'atomes, mais pour des raisons pratiques, dans une application, il peut être utile de définir une molécule comme la composition d'atomes (la gestion de leur cycle de vie en est alors bien simplifiée).

- L'association :
 - Il s'agit d'un simple lien établissant l'existence d'une relation sans subordination conceptuelle
 - ex : une personne est le frère d'une autre, un individu est le client d'un magasin



- **Le diagramme d'états-transitions :**
 - décrit les changements d'états d'un objet consécutifs aux interactions faites avec le système
 - un état est l'union des valeurs stables des attributs entre deux interactions
- Un tel diagramme est généralement un automate temporisé : ie. automate fini dont les transitions sont agrémentées de contraintes temporelles



- **Le diagramme de séquence :**
 - décrit des interactions entre objets d'un point de vue temporel
 - le point important est l'aspect communication

