

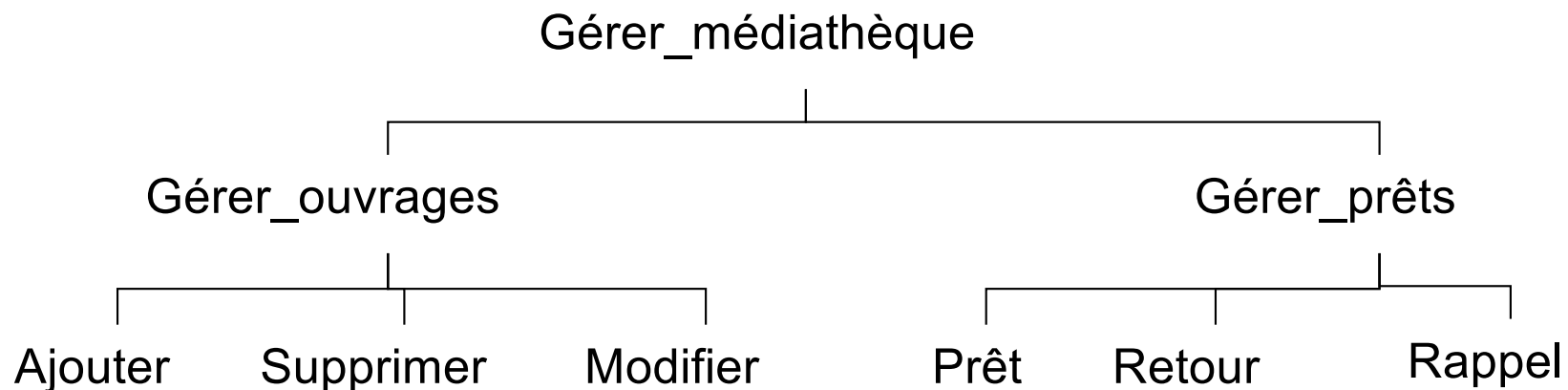
# SysML/UML

## Généralités

# Genèse d'UML : des méthodes fonctionnelles aux méthodes objet

2

- Découpe fonctionnelle : la plus intuitive :

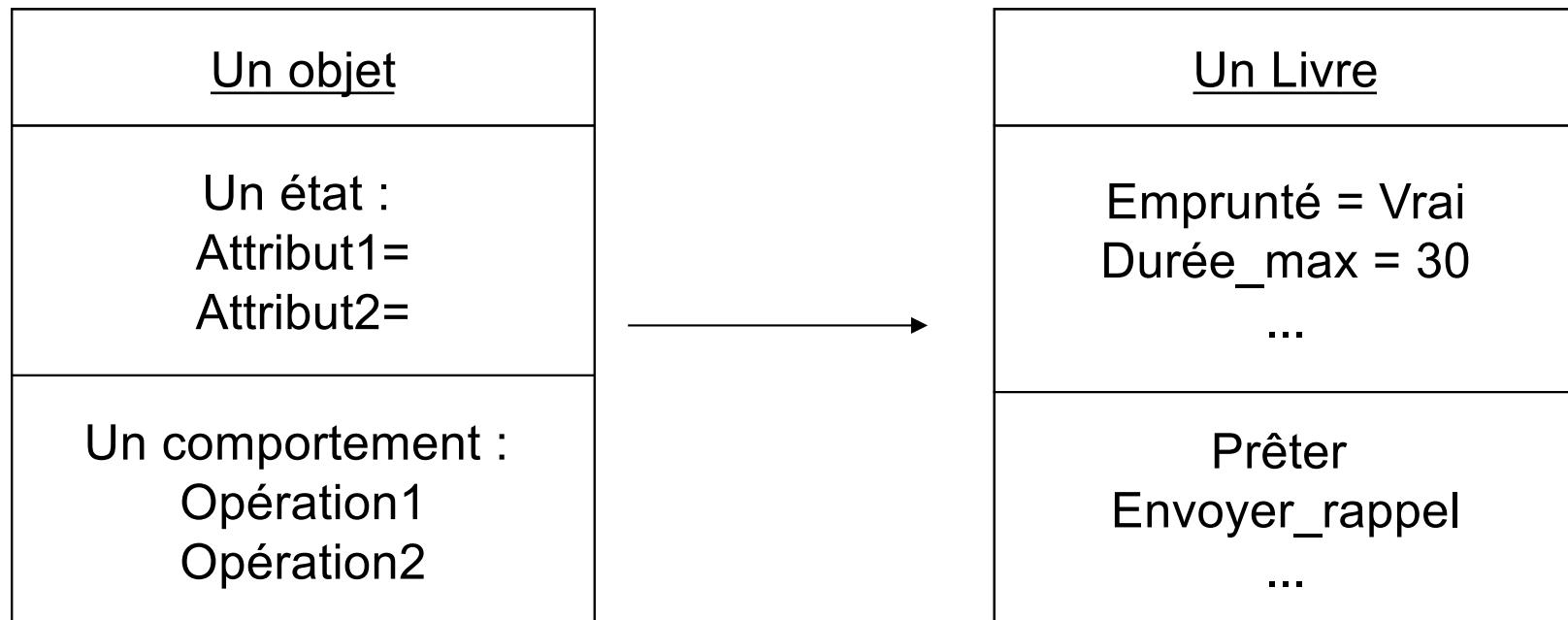


Mais l'ajout d'un nouveau type de données (K7, CD...) aux types existants (livre...) peut amener à modifier de très nombreuses fonctions

# Genèse d'UML : des méthodes fonctionnelles aux méthodes objet

3

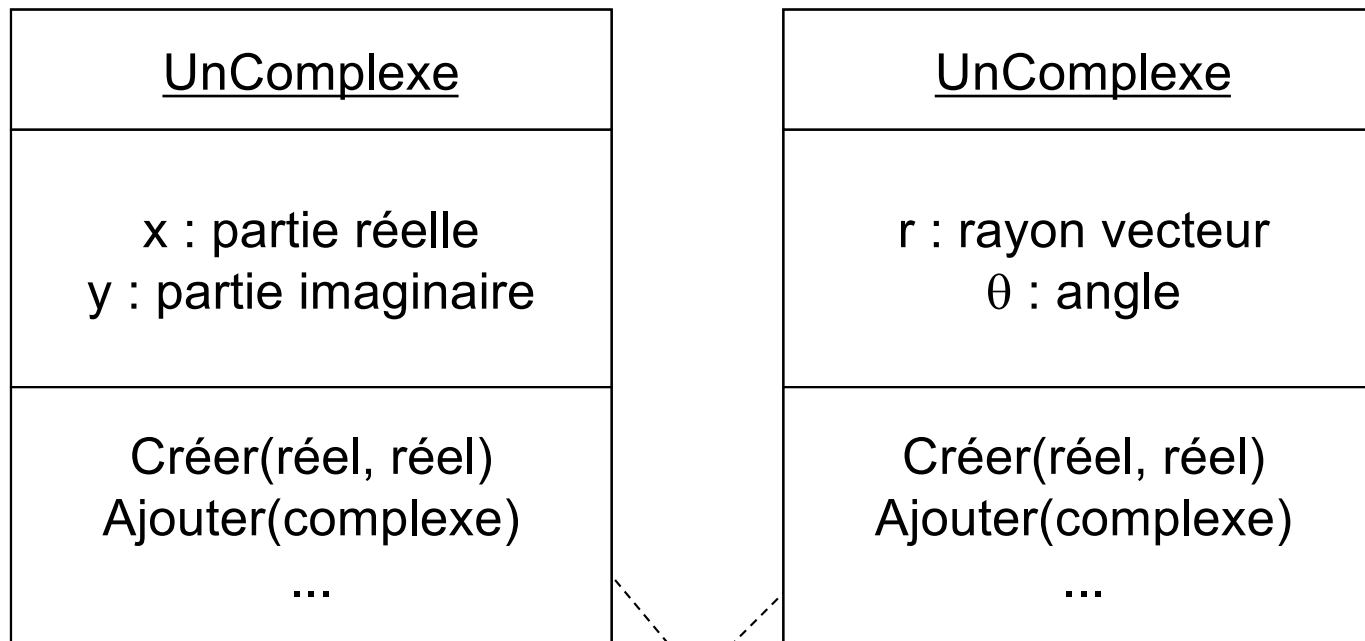
- Un objet : regroupement de l'état et du comportement dans une entité unique



# Intérêt de l'approche objet

4

- Encapsulation => réutilisabilité, évolutivité

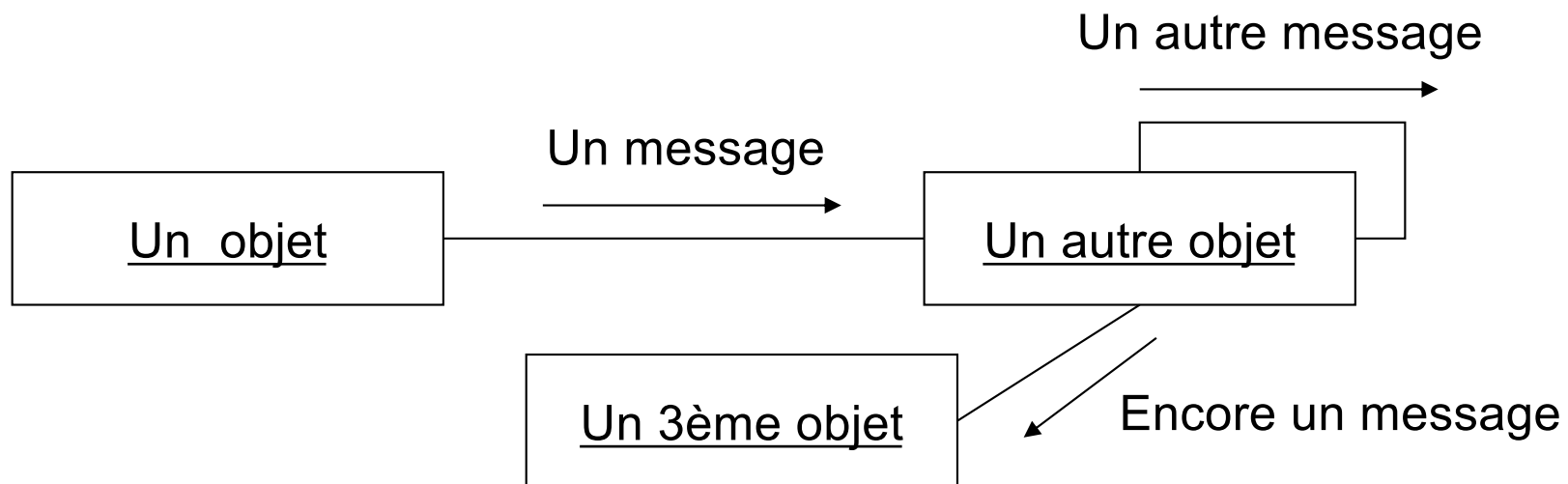


En général seules les opérations sont visibles de l'extérieur.  
La représentation interne n'est pas directement visible.

# Structure d'une application objet

5

- Application objet : ensemble d'objets collaborant à réaliser les fonctionnalités
- Les objets communiquent donc en échangeant des messages



# Genèse d'UML : Historique des méthodes objet

## 6

- Premiers langages objet (Smalltalk, C++) : début des années 1980
- Mode de pensée moins intuitif que le fonctionnel => nécessité de méthodes couvrant tout le développement (penser objet dès l'expression des besoins)
- Début des années 1990 : prolifération des méthodes objet (plus de 50 !)

# Genèse d'UML : Historique des méthodes objet

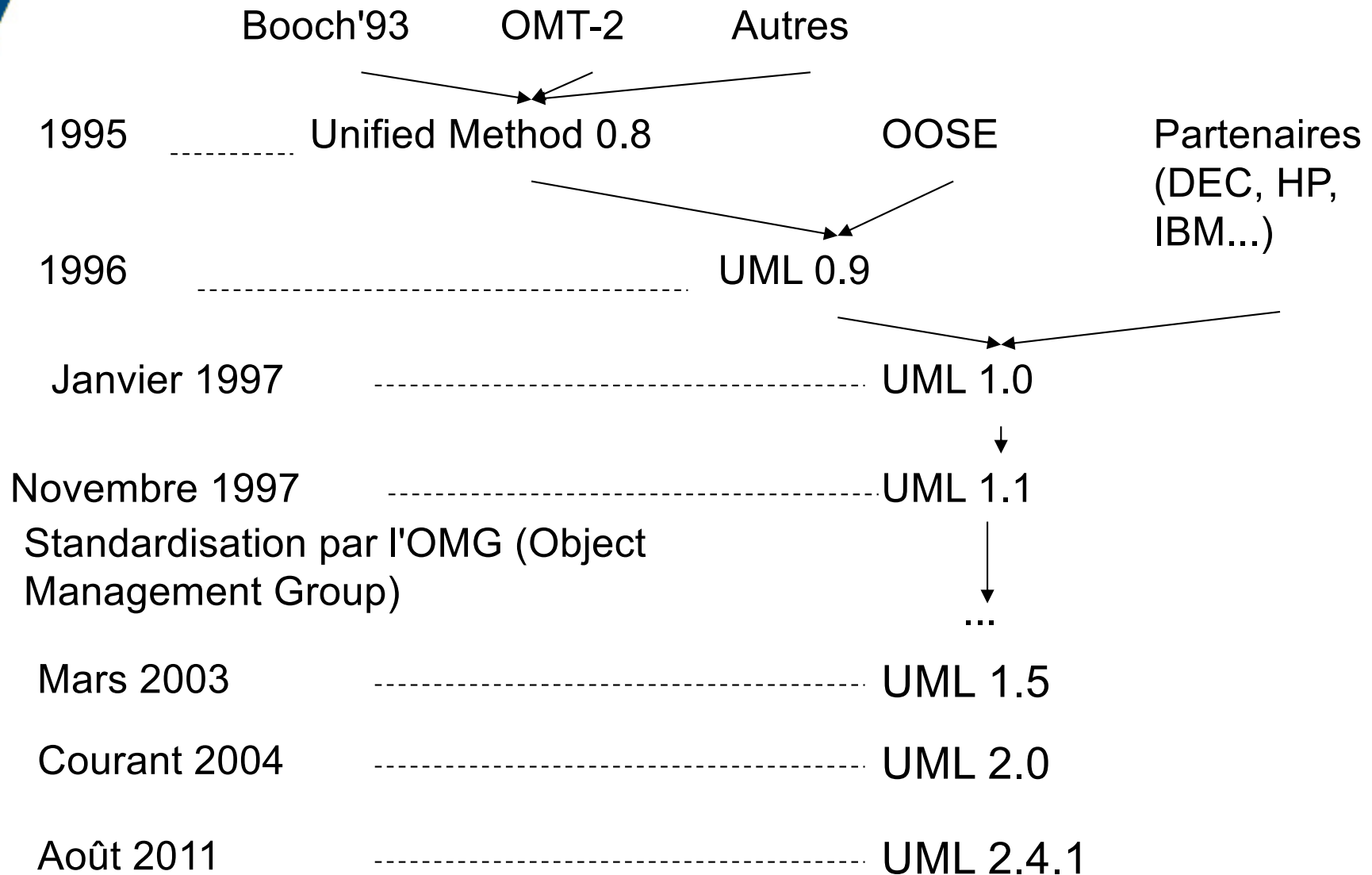
7

Les grands ancêtres :

- **OMT** : Object Modeling Technique (James Rumbaugh) : pour General Electric
- **OOD** : Object Oriented Design (Grady Booch) : pour le DOD
- **OOSE** : Object Oriented Software Engineering (Ivar Jacobson) : pour Ericsson

# Genèse d'UML : Historique des méthodes objet

## 8 • L'arbre généalogique :





# UML aujourd'hui : un standard incontournable

9

- Normalisé par l'OMG (fédère de très nombreux acteurs du monde informatique)
- Unifie le foisonnement méthodologique des années 1990
- Fait donc l'objet d'un très large consensus
- Couvre tout le cycle de vie
- Les outils UML se multiplient (Rational Rose, Rhapsody...)

# Références : Les ouvrages des pères d'UML

10

- The Unified Software Development Process, I. Jacobson, G. Booch, J. Rumbaugh, The Object Technologies Series, Addison-Wesley 1999
- The Unified Modeling Language Reference Manual, J. Rumbaugh, I. Jacobson, G. Booch, The Object Technologies Series, Addison-Wesley 1999
- The Unified Modeling Language User Guide, G. Booch, J. Rumbaugh, I. Jacobson, The Object Technologies Series, Addison-Wesley 1999

# Références : autres ouvrages et site web

11

- Modélisation objet avec UML, P.A. Muller, N. Gaertner, Eyrolles 2000 (la référence en Français)
- UML 2.0, M. Fowler, Campus Press 2004 (traduction Française de « UML distilled »)
- The Rational Unified process, P. Krutchen, The Object Technologies Series, Addison-Wesley 1999
- UML2 en action, P. Roques, F. Vallée Eyrolles 2004
- UML2 par la pratique, P. Roques Eyrolles 2004
- <http://www.rational.com> (Outil UML : Rose)
- <http://www.ilogix.com> (Outil UML : Rhapsody)
- <http://uml.free.fr>

# Qu'est-ce qu'UML ?

12

- Une notation normalisée permettant d'exprimer de manière plus ou moins formelle différentes vues d'un système :
  - Vues structurelles ou statiques
  - Vues comportementales ou dynamiques
  - Vues de l'architecture physique

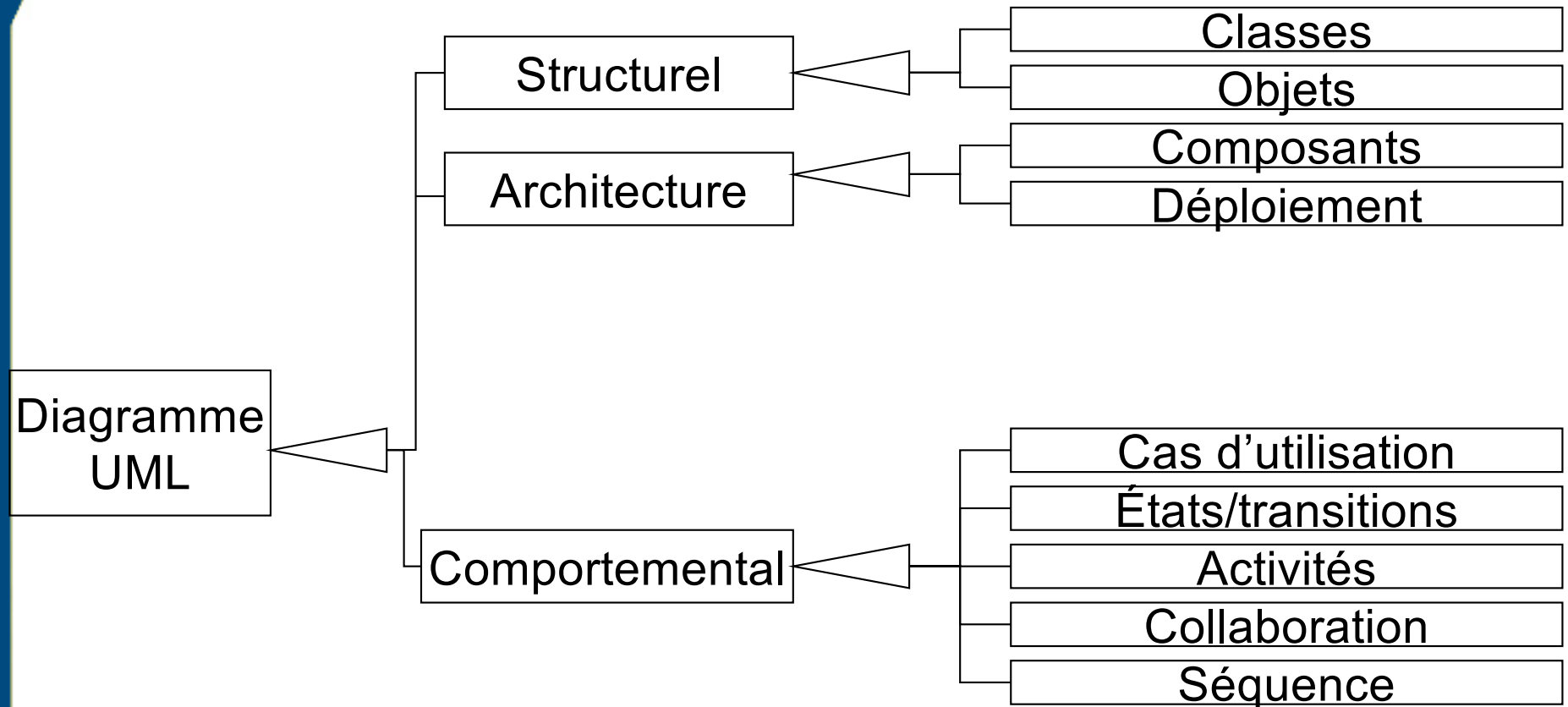
# Qu'est-ce qu'UML ?

13

- Un support de communication permettant de limiter les ambiguïtés et incompréhensions :
- UML est suffisamment formel pour :
  - se décrire lui-même (métamodèle)
  - permettre une automatisation partielle du codage
- UML n'est pas à proprement parler une méthode (ne définit pas le processus)

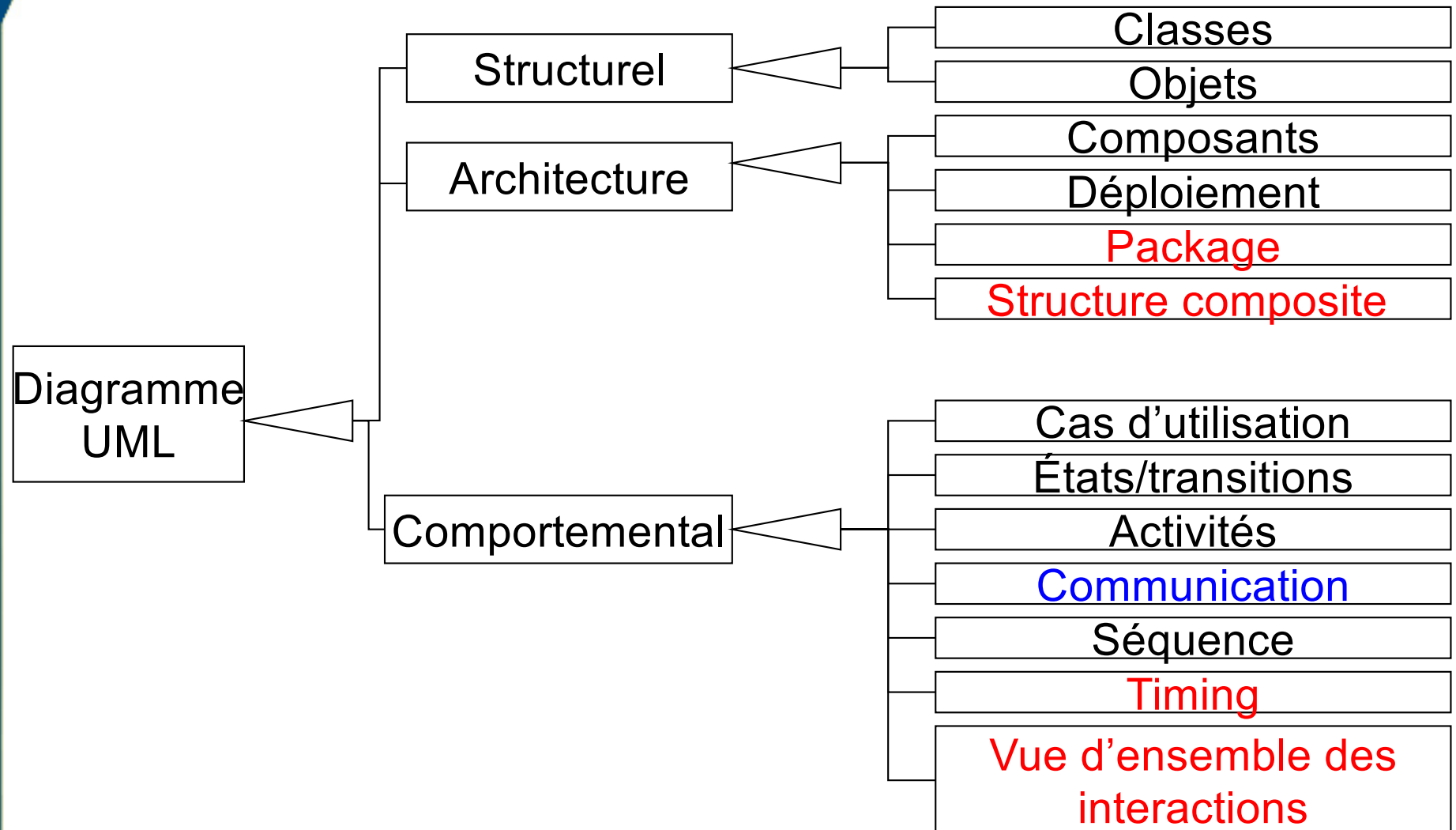
# Les diagrammes UML (-> 1.5) : 9 types

14



# Les diagrammes UML 2.0 : 13 types

15



# Les diagrammes UML

16

- UML n'est pas rigide (il est légalement admis de mixer les types de diagramme voire en ajouter d'autres).
- Personne n'utilise (voire ne comprend !) la totalité d'UML : il faut trouver le sous-ensemble qui vous est utile.
- Les types de diagrammes sont plus ou moins couramment utilisés et plus ou moins utiles. En pratique diagrammes de classes, de séquences et de cas d'utilisation constituent le sous-ensemble de base le plus utile.