



UNIVERSITÉ
DE MONTPELLIER



Langage UML

Ressource R3.03 — Analyse

Nadjib Lazaar (nadjib.lazaar@umontpellier.fr)

Modélisation

Modèle

Modélisation

Modèle

- Une **représentation**, une **généralisation**, une **abstraction** d'une réalité
 - Une représentation un sujet d'étude
 - Qui peut s'appliquer à d'autres cas d'étude
 - Incarnant un point de vue

Modélisation

Modèle

- Une **représentation**, une **généralisation**, une **abstraction** d'une réalité
 - Une représentation un sujet d'étude
 - Qui peut s'appliquer à d'autres cas d'étude
 - Incarnant un point de vue
- Modéliser pour mieux comprendre un phénomène ou mieux construire une solution

Modélisation

Modèle

- Une **représentation**, une **généralisation**, une **abstraction** d'une réalité
 - Une représentation un sujet d'étude
 - Qui peut s'appliquer à d'autres cas d'étude
 - Incarnant un point de vue
- Modéliser pour mieux comprendre un phénomène ou mieux construire une solution
- Un cas d'étude peut avoir plusieurs modèles

Modélisation

Modèle

- Une **représentation**, une **généralisation**, une **abstraction** d'une réalité
 - Une représentation un sujet d'étude
 - Qui peut s'appliquer à d'autres cas d'étude
 - Incarnant un point de vue
- Modéliser pour mieux comprendre un phénomène ou mieux construire une solution
- Un cas d'étude peut avoir plusieurs modèles
- En génie logiciel : **modélisation = spécification + conception**

Modélisation

Langage de modélisation

Modélisation

Langage de modélisation

- Langages informels
 - Langage naturel, évolution hors théorie

Modélisation

Langage de modélisation

- **Langages informels**
 - Langage naturel, évolution hors théorie
- **Langages formels**
 - Syntaxe définie par une grammaire, évolution sous le contrôle d'une théorie

Modélisation

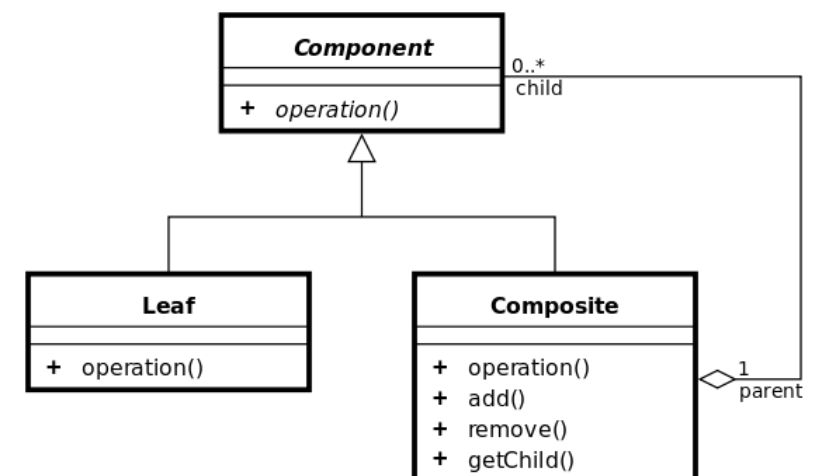
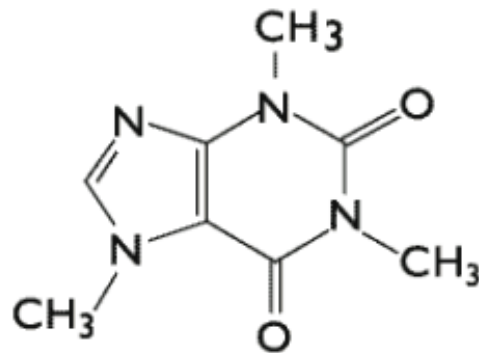
Langage de modélisation

- **Langages informels**
 - Langage naturel, évolution hors théorie
- **Langages formels**
 - Syntaxe définie par une grammaire, évolution sous le contrôle d'une théorie
- **Langages semi-formels**
 - Langage graphique (« un beau dessin vaut mieux qu'un long discours »)

Modélisation

Langage de modélisation

- **Langages informels**
 - Langage naturel, évolution hors théorie
- **Langages formels**
 - Syntaxe définie par une grammaire, évolution sous le contrôle d'une théorie
- **Langages semi-formels**
 - Langage graphique (« un beau dessin vaut mieux qu'un long discours »)



UML

Unified Modeling Language (Langage Unifié de Modélisation)

UML

Unified Modeling Language (Langage Unifié de Modélisation)

- Langage :
 - Graphique destiné à la modélisation de systèmes et de processus
 - Basé sur l'approche par objets

UML

Unified Modeling Language (Langage Unifié de Modélisation)

- Langage :
 - Graphique destiné à la modélisation de systèmes et de processus
 - Basé sur l'approche par objets
- Unifié :
 - Provient de plusieurs notations qui l'ont précédé
 - Promu par OMG (Object Management Group)

UML

Unified Modeling Language (Langage Unifié de Modélisation)

- **Langage :**
 - Graphique destiné à la modélisation de systèmes et de processus
 - Basé sur l'approche par objets
- **Unifié :**
 - Provient de plusieurs notations qui l'ont précédé
 - Promu par OMG (Object Management Group)
- **Modélisation :**
 - Grande richesse sémantique qui permet l'abstraction de nombreux aspects techniques

UML

Unified Modeling Language (Langage Unifié de Modélisation)

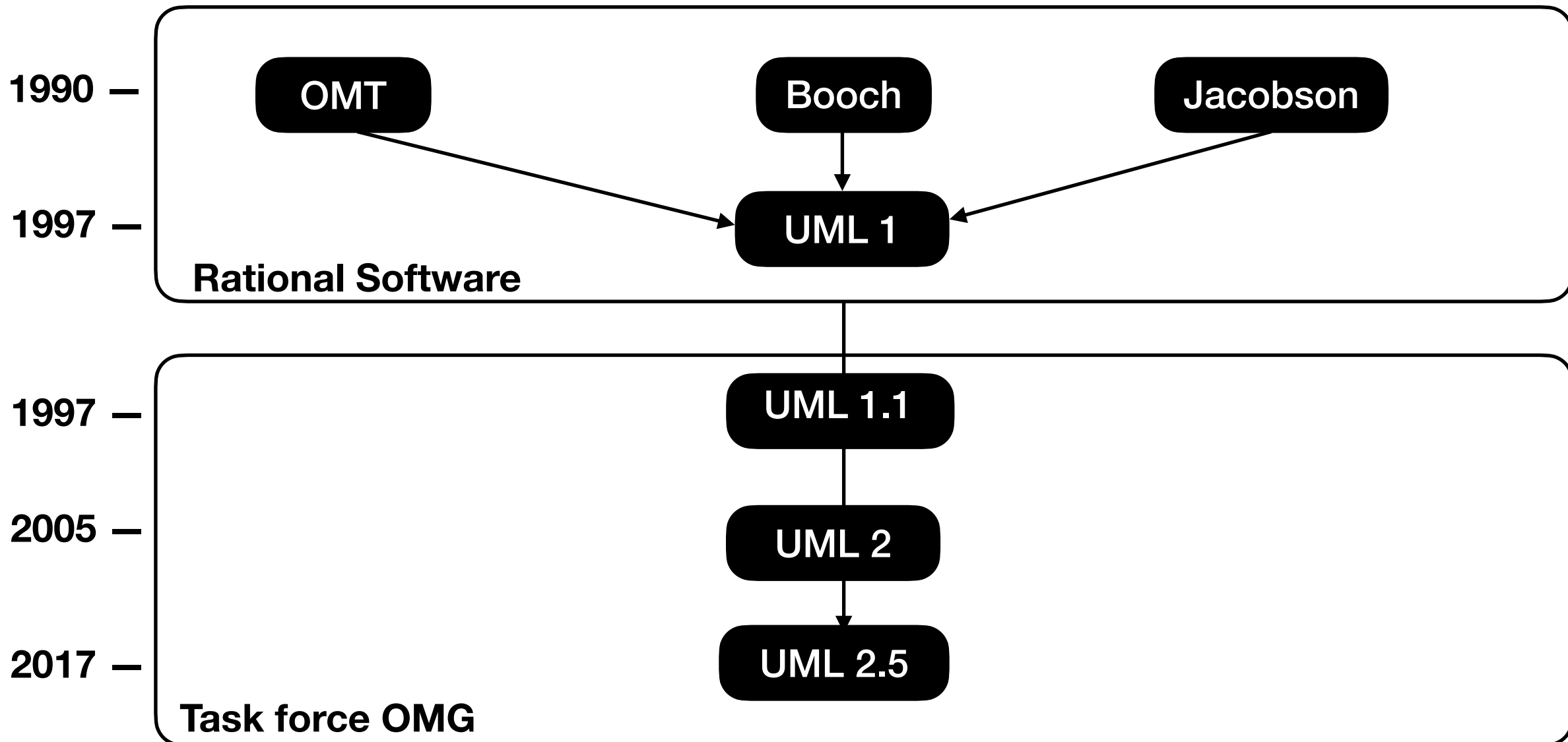
- Langage :
 - Graphique destiné à la modélisation de systèmes et de processus
 - Basé sur l'approche par objets
- Unifié :
 - Provient de plusieurs notations qui l'ont précédé
 - Promu par OMG (Object Management Group)
- Modélisation :
 - Grande richesse sémantique qui permet l'abstraction de nombreux aspects techniques



UML est un langage, pas une méthode !

UML

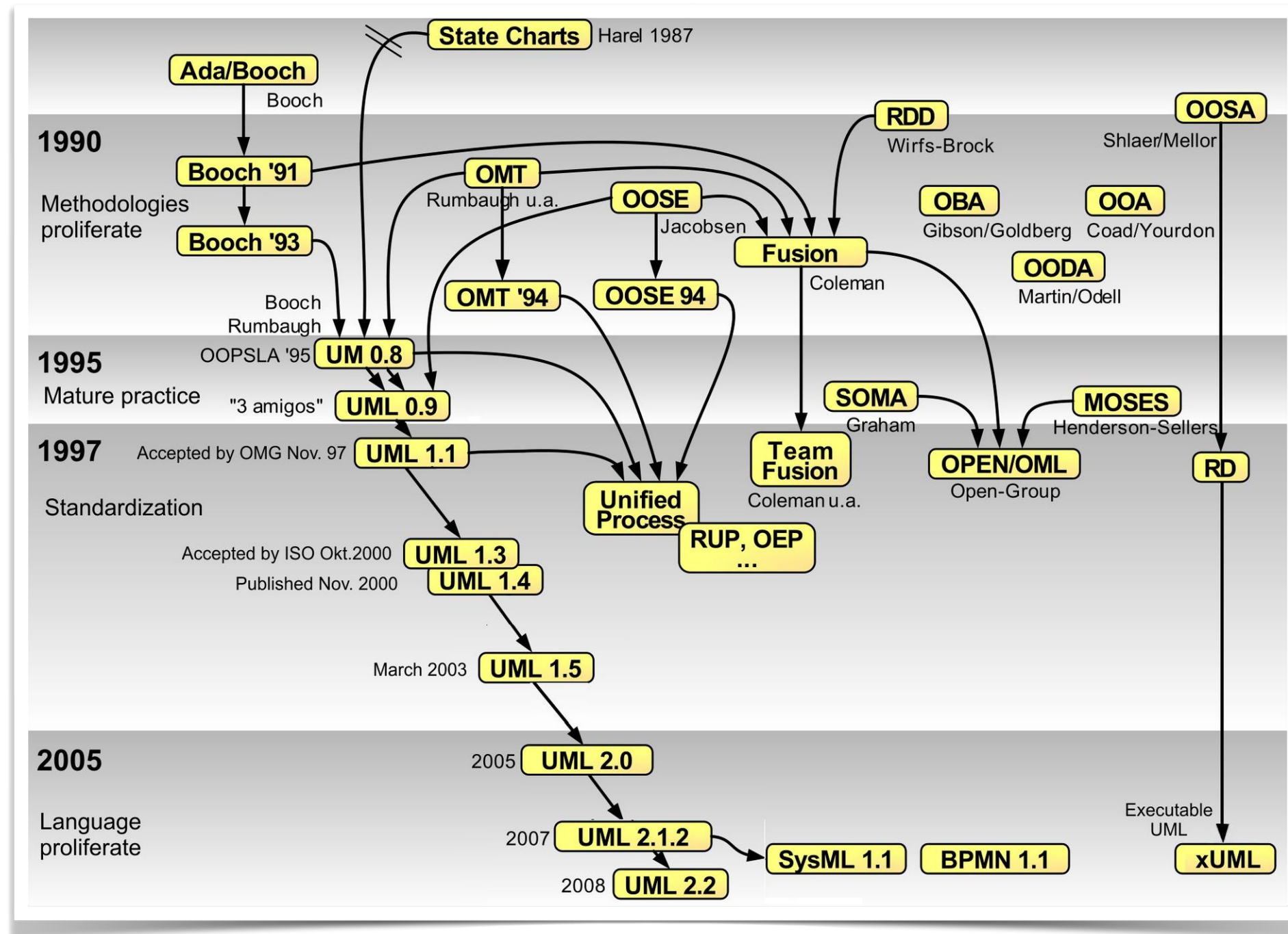
Historique



Dernière version : UML 2.5.1 (12/2017)

UML

Historique complet (wiki)



UML

Pourquoi UML ?

UML

Pourquoi UML ?

- Besoin d'une modélisation pour construire un logiciel

UML

Pourquoi UML ?

- Besoin d'une modélisation pour construire un logiciel
- Besoin de modéliser les aspects statiques et les aspects dynamiques

UML

Pourquoi UML ?

- Besoin d'une modélisation pour construire un logiciel
- Besoin de modéliser les aspects statiques et les aspects dynamiques
- Besoin de modéliser différents point de vue

UML

Pourquoi UML ?

- Besoin d'une modélisation pour construire un logiciel
- Besoin de modéliser les aspects statiques et les aspects dynamiques
- Besoin de modéliser différents point de vue
- Indépendant du processus du développement et des langages de programmation

UML

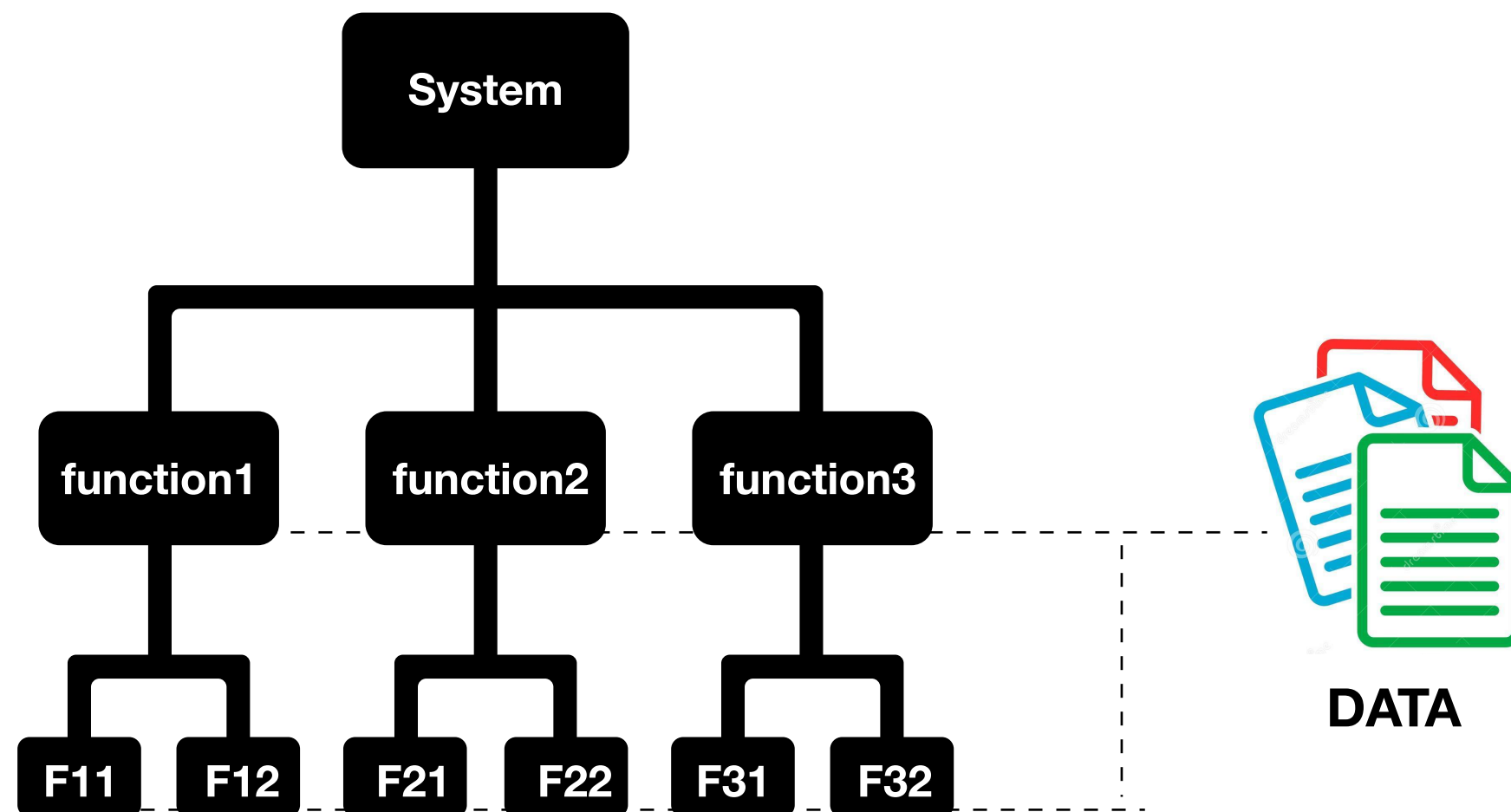
Pourquoi UML ?

- Besoin d'une modélisation pour construire un logiciel
- Besoin de modéliser les aspects statiques et les aspects dynamiques
- Besoin de modéliser différents point de vue
- Indépendant du processus du développement et des langages de programmation
- Un outil efficace de penser le logiciel

Conception

Méthodes

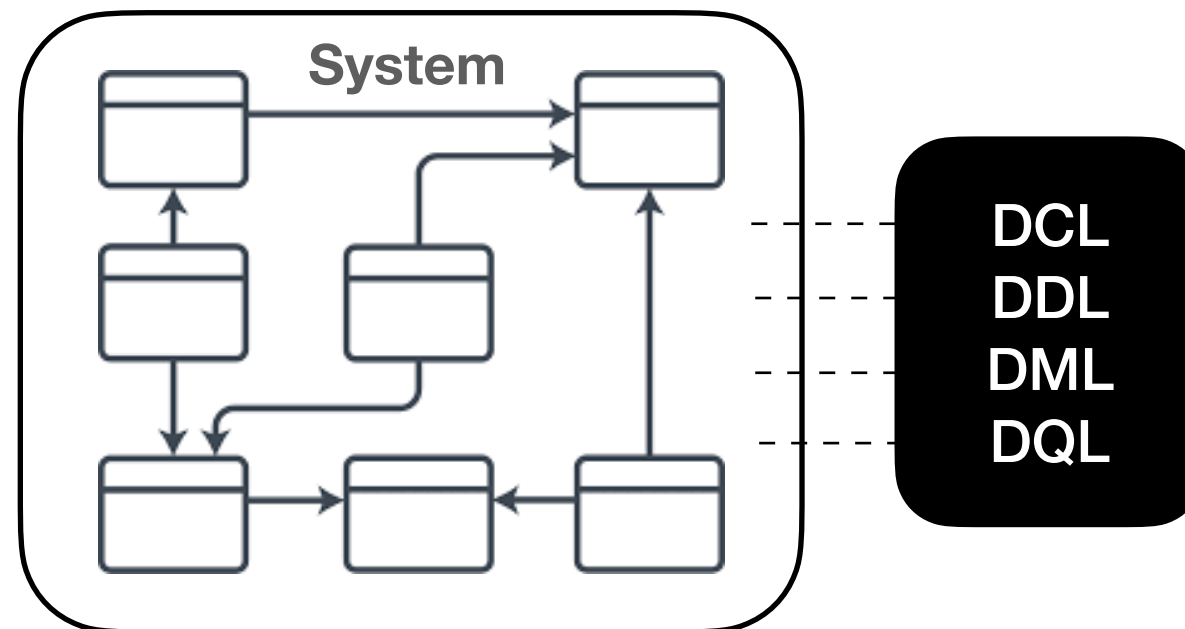
- Conception fonctionnelle



Conception

Méthodes

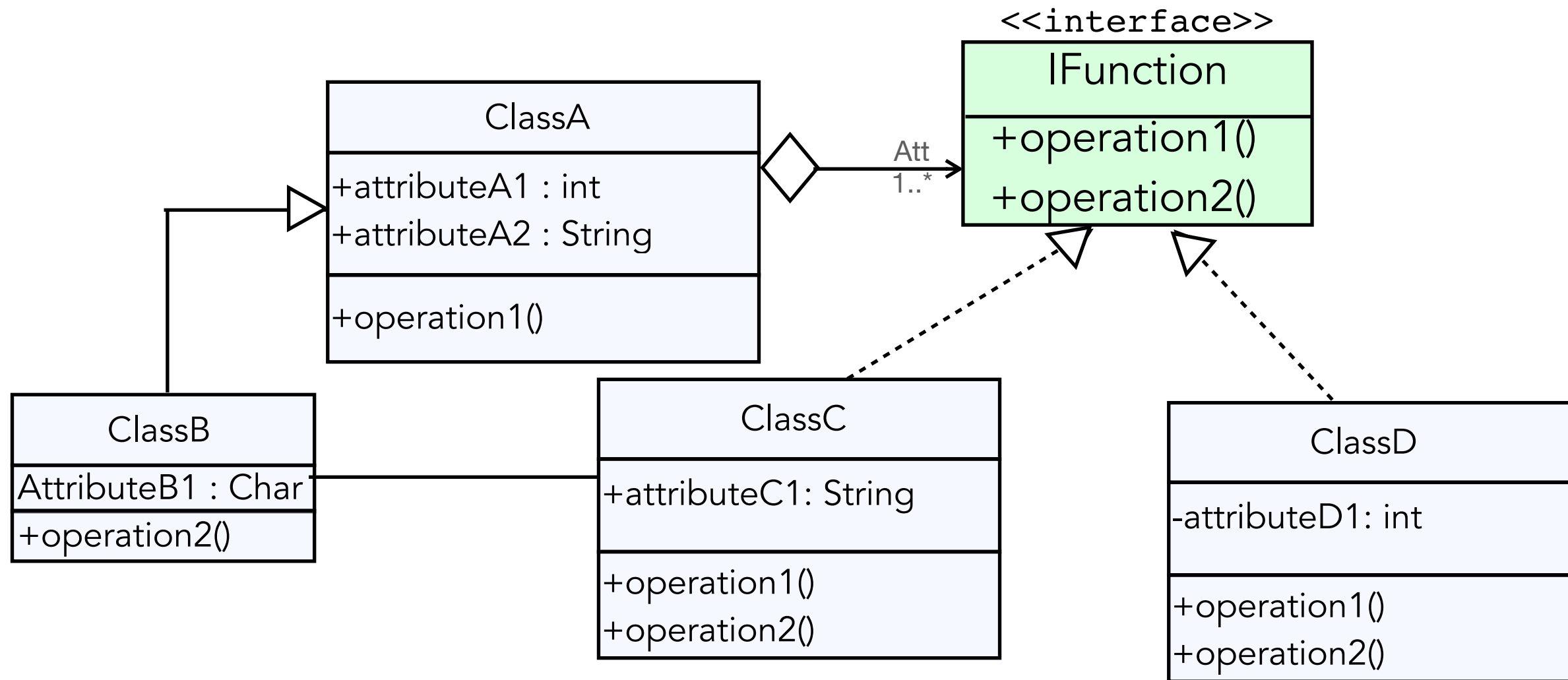
- Conception guidée par les données



Conception

Méthodes

- Conception Orientée Objet

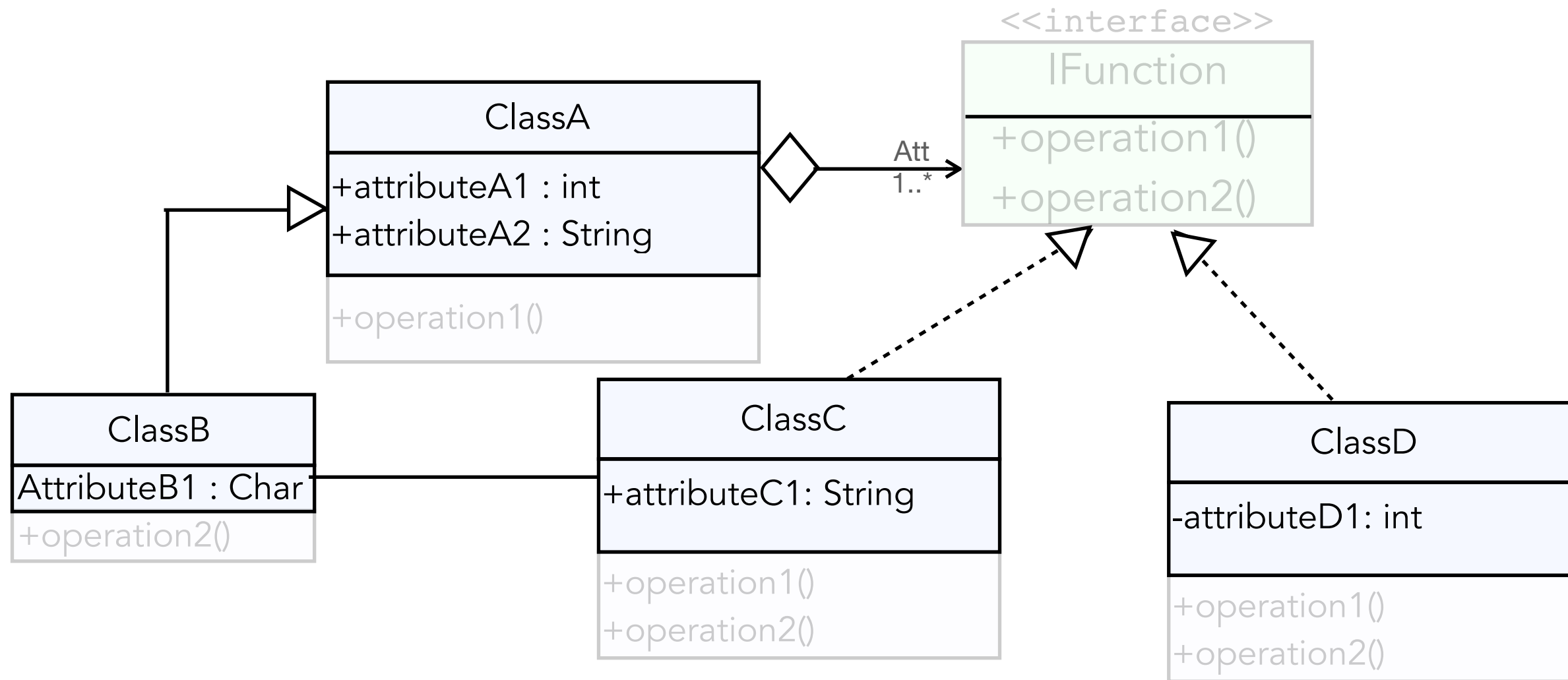


Systeme = ensemble d'objets

Conception

Méthodes

- Conception Orientée Objet

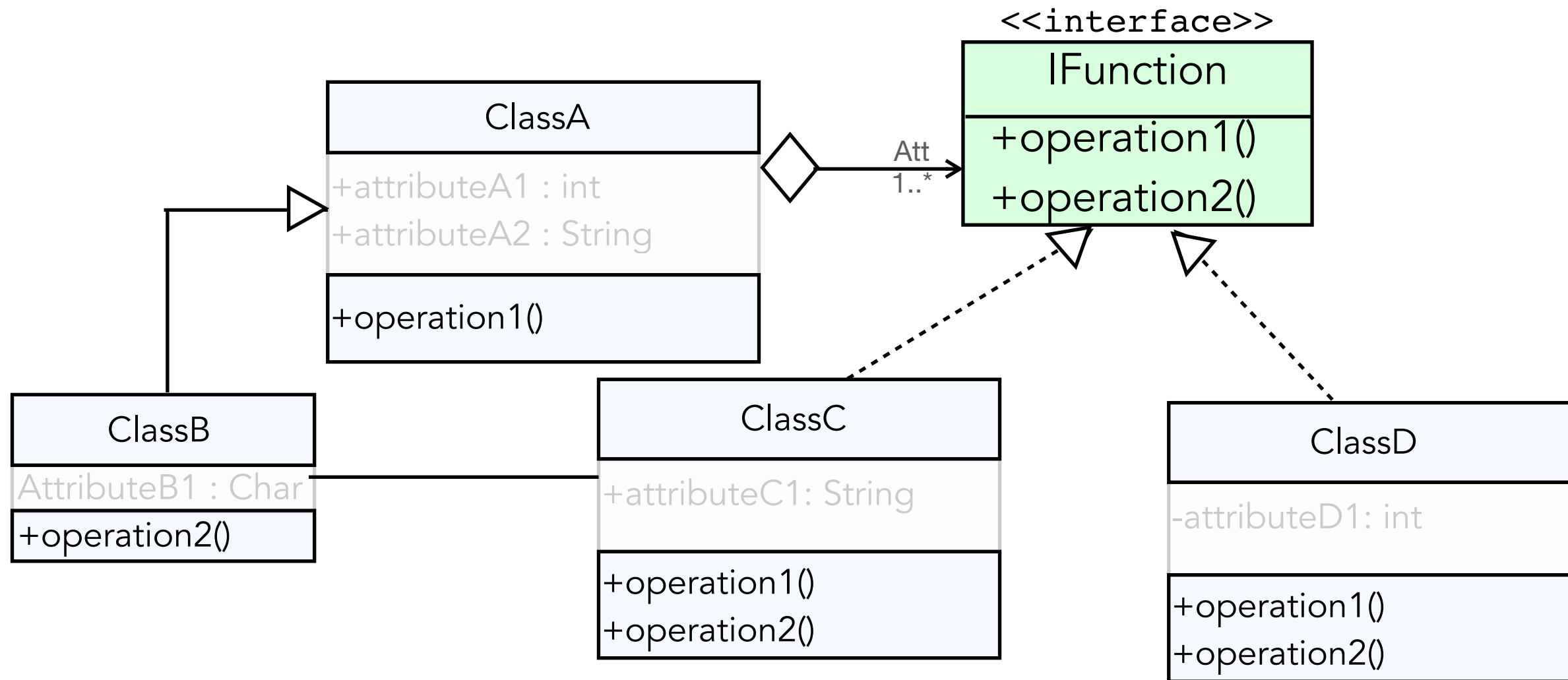


Objet = **Données** + fonctions

Conception

Méthodes

- Conception Orientée Objet



Objet = Données + **fonctions**

UML

Utilisation

UML

Utilisation

- **Trois modes d'emploi (Ways of Using UML [Fowler2004]) :**

UML

Utilisation

- **Trois modes d'emploi (Ways of Using UML [Fowler2004]) :**
 - **Mode esquisse (UML as sketch)** : des diagrammes sur papier pour améliorer la communication

UML

Utilisation

- **Trois modes d'emploi (Ways of Using UML [Fowler2004]) :**
 - **Mode esquisse (UML as sketch)** : des diagrammes sur papier pour améliorer la communication
 - **Mode plan (UML as blueprint)** : des diagrammes d'analystes/concepteurs pour des programmeurs

UML

Utilisation

- **Trois modes d'emploi (Ways of Using UML [Fowler2004]) :**
 - **Mode esquisse (UML as sketch)** : des diagrammes sur papier pour améliorer la communication
 - **Mode plan (UML as blueprint)** : des diagrammes d'analystes/concepteurs pour des programmeurs
 - **Mode exécutable (UML as Programming language)** : Des solutions UML compilable et exécutables

UML

Architecture guidée par les modèles (MDA)

- **Model-Driven Architecture : proposition de l'OMG**
- **Modélisation en faisant abstraction des aspects technologiques**

UML

Architecture guidée par les modèles (MDA)

- **Model-Driven Architecture : proposition de l'OMG**
- **Modélisation en faisant abstraction des aspects technologiques**

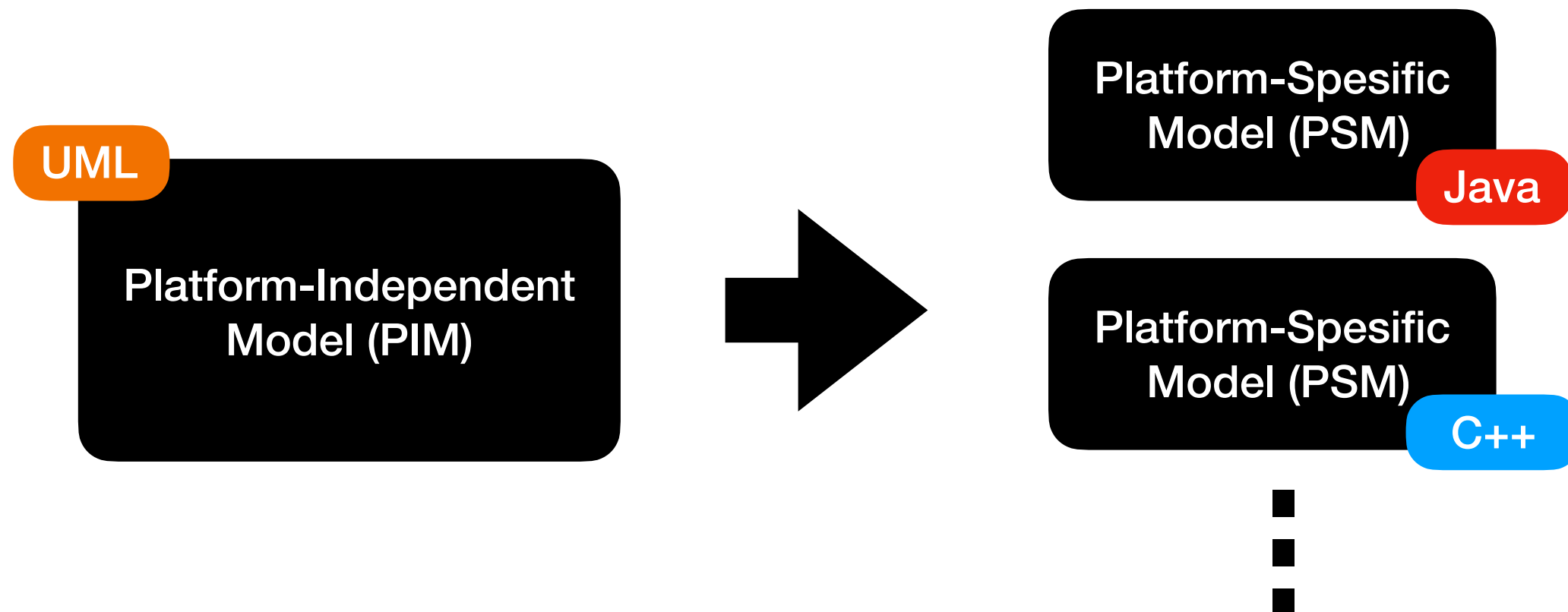
UML

Platform-Independent
Model (PIM)

UML

Architecture guidée par les modèles (MDA)

- **Model-Driven Architecture : proposition de l'OMG**
- **Modélisation en faisant abstraction des aspects technologiques**



UML

Les diagrammes

UML

Les diagrammes

- **Diagrammes descriptifs / prescriptifs**
 - Décrire un existant
 - Décrire un système à réaliser

UML

Les diagrammes

- **Diagrammes descriptifs / prescriptifs**
 - Décrire un existant
 - Décrire un système à réaliser
- **Moyen de communication entre les différents acteurs :**
 - Client / Analyste / Concepteur / Programmeur / Testeur / Formateur

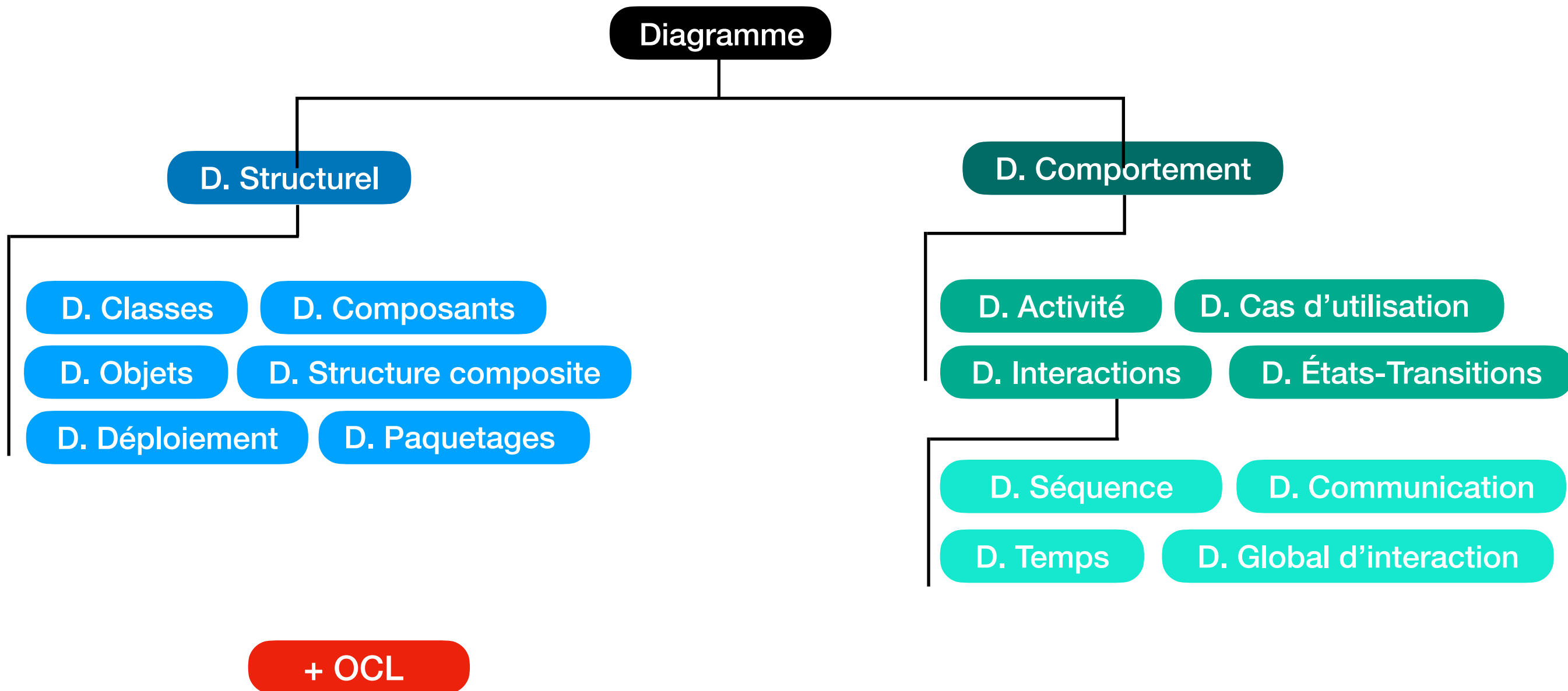
UML

Les diagrammes

- **Diagrammes descriptifs / prescriptifs**
 - Décrire un existant
 - Décrire un système à réaliser
- **Moyen de communication entre les différents acteurs :**
 - Client / Analyste / Concepteur / Programmeur / Testeur / Formateur
- **Description des aspects statiques / dynamiques du système**
 - Aspects structurels
 - Comportements et interactions

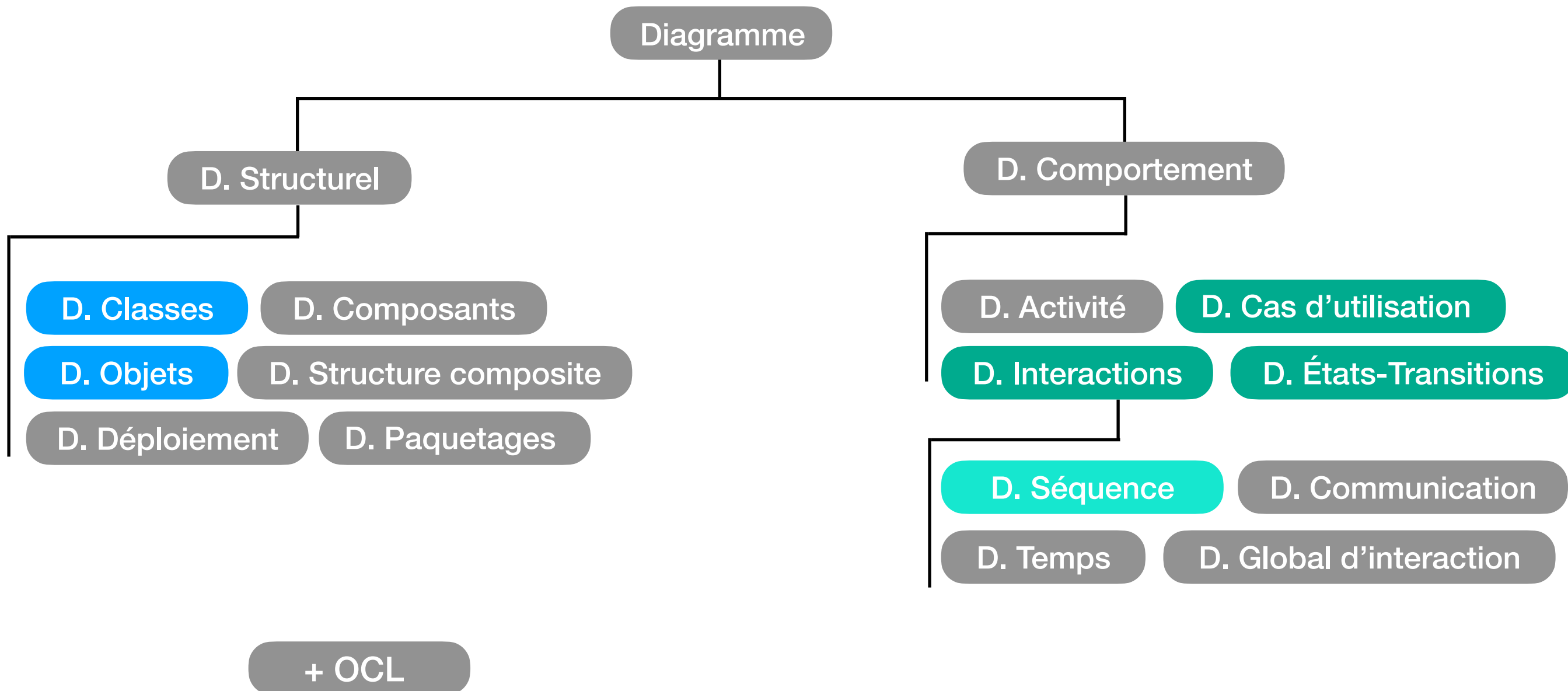
UML

Les diagrammes



UML

Les diagrammes



UML

Dans ce cours

- **Specification**

- Besoins des utilisateurs (diag. cas d'utilisations)
- Interaction Utilisateur <-> Logiciel (diag. séquence)

- **Conception**

- Structure interne du logiciel (diag. classes)
- État interne du logiciel à l'instant T (diag. objets)
- Évolution des objets (diag. états-transitions)
- Interaction des objets (diag. séquence)

Références

Books

- **UML Distilled (Third Edition): A Brief Guide to the Standard Object Modeling Language.** M Fowler 2004.
- **Object-Oriented Software Engineering (Second Edition): Practical Software Development Using UML and Java.** T. Lethbridge and R. Laganière 2005.
- **UML in Practice: The Art of Modeling Software Systems Demonstrated through Worked P.** Rogues 2004.
- **Requirements Engineering: From System Goals to UML Models to Software Specifications.** A. Lamsweerde 2009.
- **Software Engineering with UML.** B. Unhelkar 2018.

Many

Thanks to

Noureddine Aribi II Sébastien Bardin II Nassim Belmecheri II
Kevin Chapuis II Pierre Gérard II Arnaud Gotlieb II Marianne Huchard II
Laurent Jean-Pierre II Alain Joubert II Yahia Lebbah II
Delphine Longuet II Samir Loudni II Sophie Nabitz II
Clémentine Nebut II Matthieu Rosenfeld II Alain Sabatier II
Helmut Simonis II Djamel Seriali II Christine Solnon II
Julie Vachon II Petru Valicov II Keunhyuk Yeom II Raphael Yende