



Modélisation UML

Aurore Blot – GRTgaz
aurore.blot@insa-rouen.fr



Chapitre 01 – Généralités

- Contexte
- Qu'est-ce qu'un modèle
- Qu'est-ce que UML ?
- Vue d'ensemble



Chapitre 01 – Généralités

- Contexte
- Modèle & Langage
- Qu'est-ce que UML ?
- Vue d'ensemble



Objectifs du cours

- Connaître le langage de modélisation UML
- Comprendre la sémantique des éléments des principaux modèles
- Utiliser des diagrammes UML pour modéliser un objet d'étude
- Concevoir l'architecture d'un logiciel orienté objet

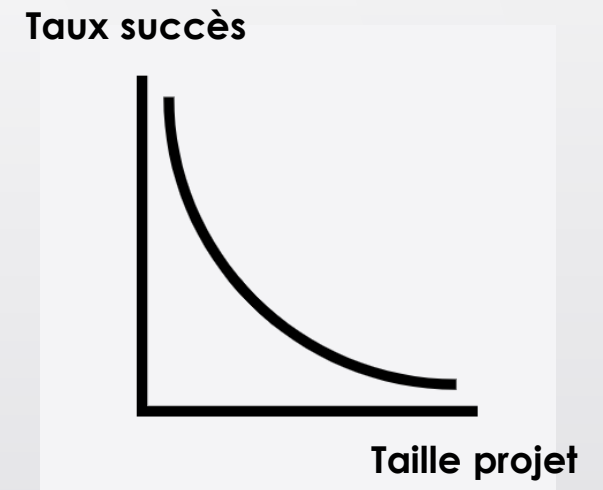
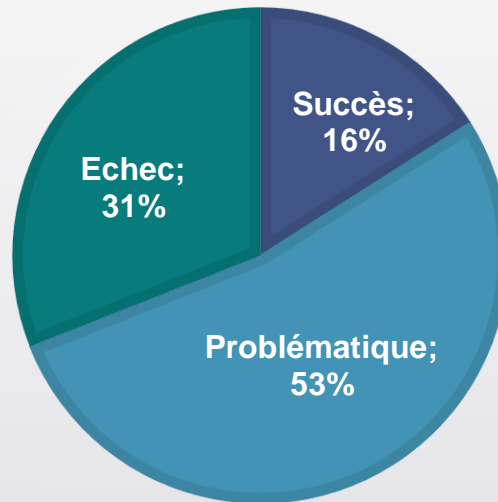


Contexte

- Quel que soit le domaine d'application, on a un prorata proche de :
 - 80% de software
 - 20% d'hardware.
- Hardware : plutôt fiable et standardisé.
- Les « surprises » viennent essentiellement du software.

La « crise du logiciel »

- Etude Standish Group (1995), sur un peu plus de 8000 projets dans l'IT



- Comment faire des logiciels de qualité ?
- Quels sont les critères de qualité ?



Qualité logicielle

- La qualité du produit final est directement liée à la qualité du processus d'élaboration.
- Processus d'élaboration = cycle de vie du logiciel

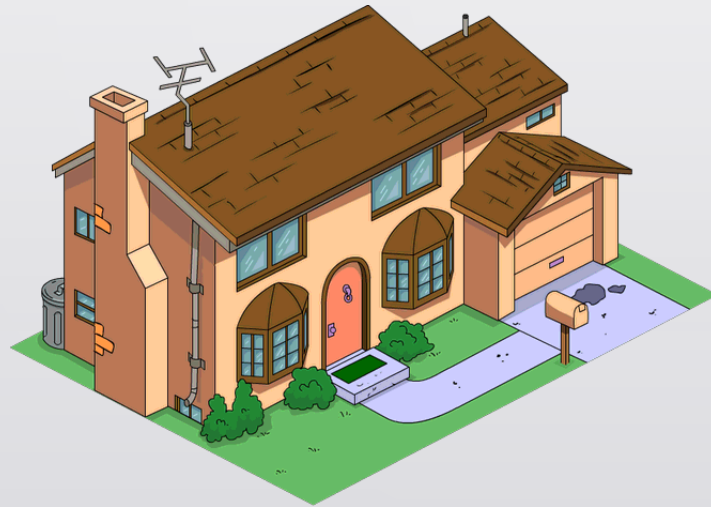


Cycle de vie d'un logiciel

- Cahier des charges
- Etude de faisabilité
- Spécifications
- Conception
- Codage
- Tests (parfois avant le codage)
- Maintenance

- Conception ~ Modélisation
- Exemple

De l'importance de la conception





Chapitre 01 – Généralités

- Contexte
- Modèle & Langage
- Qu'est-ce que UML ?
- Vue d'ensemble



De l'importance de la modélisation

- La modélisation permet de :
- Visualiser **le système tel qu'il est / devrait être**
- Spécifier la **structure** du système
- Spécifier le **comportement** du système
- **Documenter** les décisions

Grâce à la modélisation,

on appréhende des systèmes dans leur entièreté.



Qu'est-ce qu'un modèle ?

- **Modèle = représentation abstraite** de la réalité, excluant certains détails, ayant les propriétés suivantes :
 - **Représentativité**
 - **Généricité**
 - **Abstraction**
- Exemple : les circuits électriques / diverses mesures sur des circuits / $U=RI$
- Un même système peut avoir plusieurs modèles. Chacun donne un point de vue différent et complémentaire sur le système.



Rappel de modélisation orientée objet

- Modélisation par décomposition fonctionnelle
 - Approche descendante
 - La fonction donne la forme du système.
- Modélisation Orientée Objet
 - Approche principalement ascendante
 - La structure du système lui donne sa forme.



Rappel Modélisation objet

- **Unicité et universalité**

Réduire le décalage entre monde réel et logiciel

- **Réutilisabilité et évolutivité**

Encapsulation, modularité, abstraction, polymorphisme, héritage...

- **Arrive à maturité**

Bibliothèque de classes, Modélisation, Méthode de développement...



Rappel Modélisation objet

- **Objet** = Etat + Comportement + Identité
- **Classe d'objet** = regroupement d'objets similaires (instances)



Qu'est-ce qu'un langage ?

- **Langues naturelles** : Évoluent hors du contrôle d'une théorie (français, anglais, ...)
- **Langages artificiels** : Connus pour des usages particuliers (logique, langage informatique, ...)
 - Langage **formels** : le plus souvent mathématiques, permettant des preuves formelles
 - Langage **semi-formels** : le plus souvent graphique, plus faciles à utiliser
- Un langage de modélisation doit définir : une sémantique des concepts, une notation pour les représenter, des règles d'utilisation



Chapitre 01 – Généralités

- Contexte
- Modèle & Langage
- Qu'est-ce que UML ?
- Vue d'ensemble



OMG et UML

- **OMG = Object Management Group**
(www.omg.org)
 - Fondé en 1989
 - Objectif : standardiser et promouvoir l'objet
 - 1997 : 1^{ère} version d'UML
 - 2005 : UML 2.0 – version majeure (changements structurant permettant d'élargir le champ d'application d'UML)
 - 2012 : UML 2.5
- **UML = Unified Modeling Language**
 - Langage visuel dédié à la spécification, construction, documentation des artefacts d'un système logiciel
 - UML est un langage, pas une méthode !



UML Contexte

- **Dans les méthodes Agile**
 - Diagrammes informels, incomplets
 - Pour les parties critiques uniquement
 - Support de communication
- **Mode Plan**
 - Plus formel et plus détaillés
 - Squelette de code à partir des diagrammes
- **Mode Programmation**
 - Complet et formel



UML

Deux catégories

- **Modèles statiques**
 - Décrit la structure du système
- **Modèles dynamiques**
 - Décrit le comportement et les interactions



UML

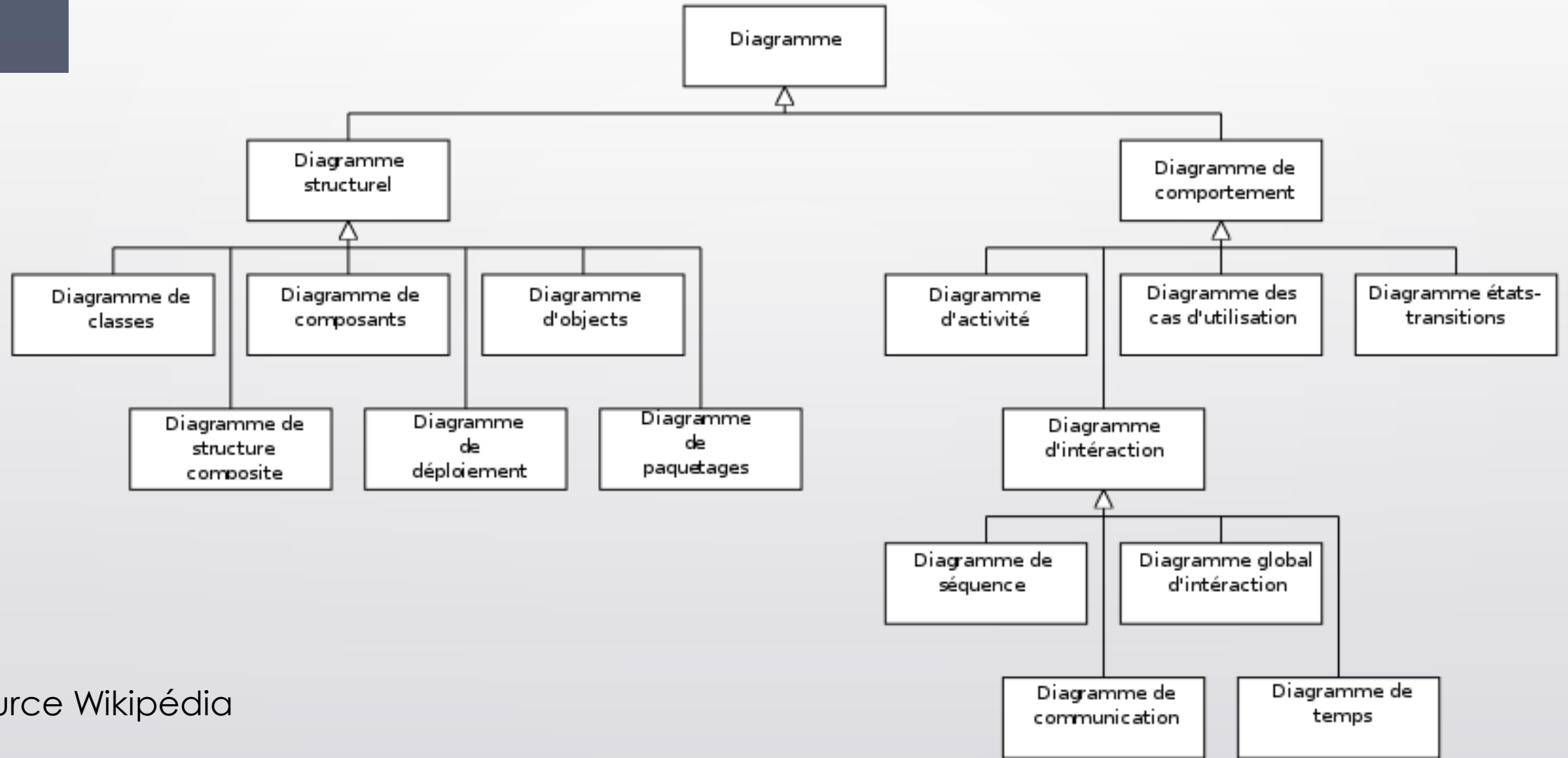
Cas d'application

- Décrire l'existant
- Décrire le futur système
- Répondre à la question « quoi? » (pour le client)
- Répondre à la question « comment » (pour les concepteurs / développeurs)

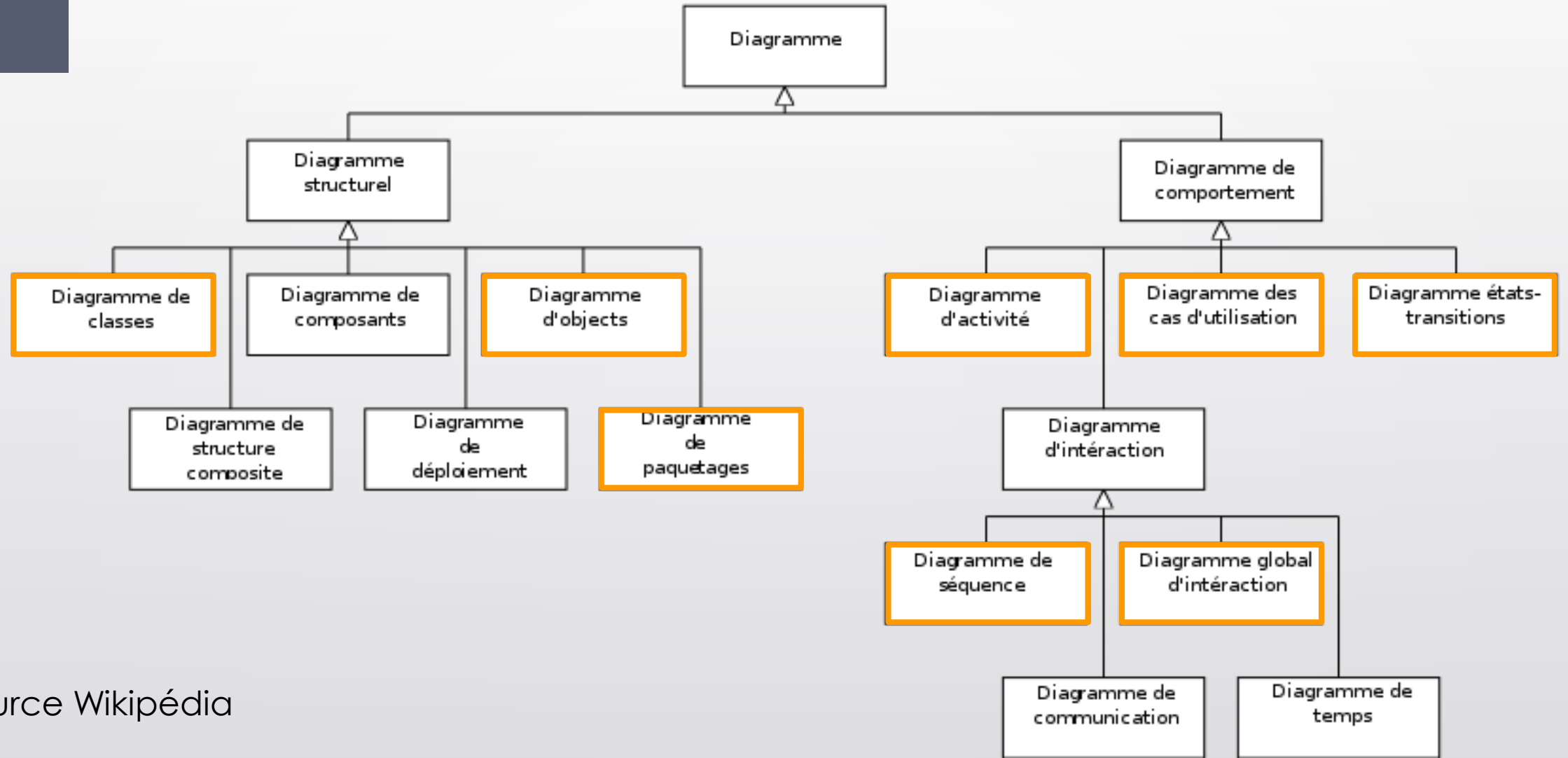


Chapitre 01 – Généralités

- Contexte
- Modèle & Langage
- Qu'est-ce que UML ?
- Vue d'ensemble



Source Wikipédia



Source Wikipédia



Chapitre 01 – Généralités

- Contexte
- Qu'est-ce qu'un modèle
- Qu'est-ce que UML ?
- Vue d'ensemble