

UML: Unified Modeling Language

Introduction

UML est la forme contractée de « Unified Modeling Language ».

UML est un langage graphique conçu pour représenter, spécifier, construire et documenter les artefacts d'un système à dominante logicielle

Pourquoi UML ?

De la même façon qu'il vaut mieux dessiner une maison avant de la construire, il vaut mieux modéliser un système avant de le réaliser.

- Obtenir une modélisation de très haut niveau indépendante des langages et des environnements.
- Faire des simulations avant de construire un système.
- Documenter un projet.
- Générer automatiquement la partie logiciel d'un système.
- Fournir aux utilisateurs un langage de modélisation visuelle expressif et prêt à l'emploi

Utilisation

La modélisation consiste à créer une représentation simplifiée d'un problème: de ce fait. le model permet de représenter simplement un problème, un concept et le simuler. La modélisation comporte deux composantes :

- L'analyse, c'est-à-dire l'étude du problème
- la conception, soit la mise au point d'une solution au problème Le modèle constitue ainsi une représentation possible du système pour un point de vue donné.

Origine

1995

- James Rumbaugh Et Grady Booch publie un document intitulé "Unified Méthode V0.8"

1996

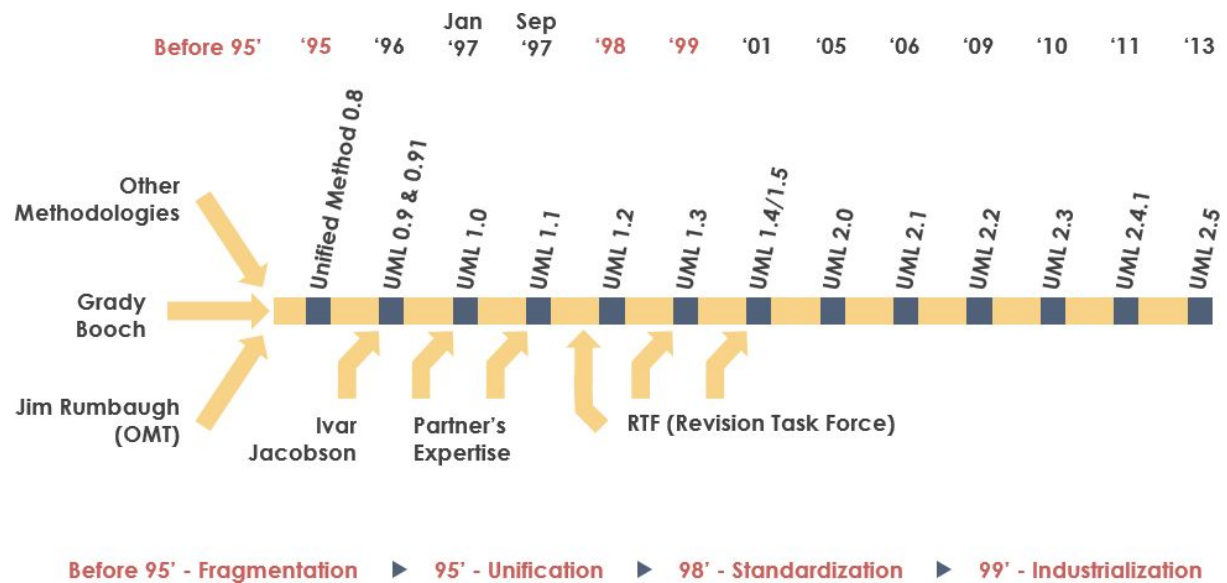
- Ivar Jacobson se joint au James et Grady et les trois publie la version 0.9
- consortium formé de grandes entreprises telles que Microsoft, IBM, Oracle, et autres est créé dans le but de faire évoluer UM vers la version 1.0.

1997

- UM change de nom pour UML (Unified Modeling Language)
- l'association Object Management Group (OMG) normalise UML 1.0

1997-2015

- UML évolue de la V1.0 jusqu'à V2.5



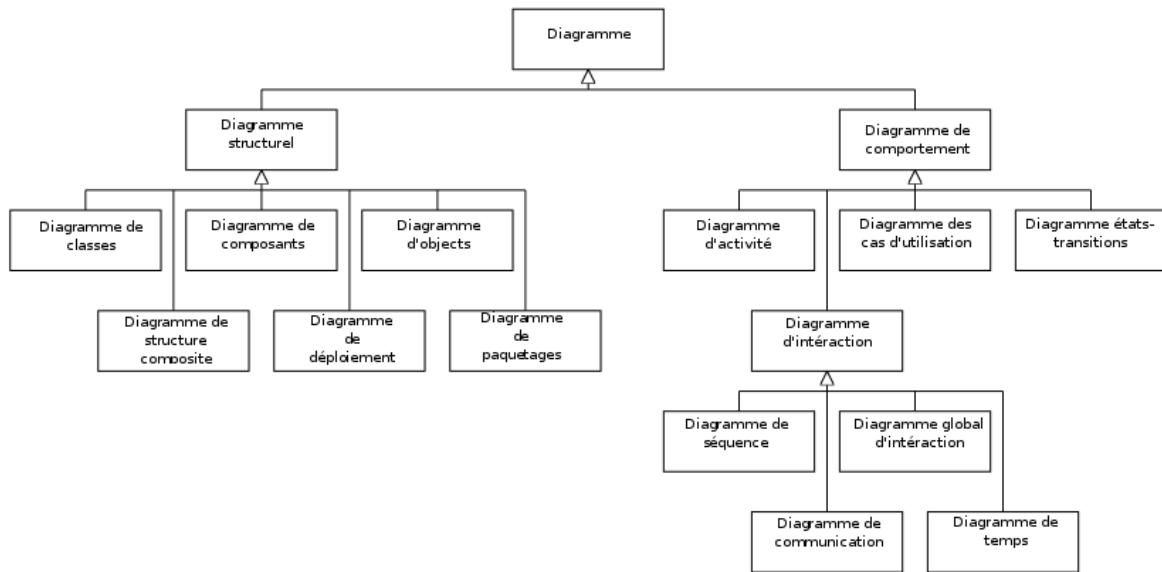
Les différents types de diagrammes

Vues statiques

- **Diagrammes de classes** : représentent la structure statique en termes de classes et de relations
- **Diagrammes d'objets** : représentent les objets et leurs relations
- **Diagrammes de cas d'utilisation** : représentent les fonctions du système du point de vue de l'utilisateur
- **Diagrammes de composants** : représentent les composants physiques d'une application
- **Diagrammes de déploiement** : représentent le déploiement des composants sur des dispositifs matériels

Vues dynamiques

- **Diagrammes de séquence** : sont une représentation temporelle des interactions entre objets
- **Diagrammes de collaboration** : sont une représentation spatiale des objets et de leurs interactions
- **Diagrammes d'états-transitions** : représentent le comportement d'une classe en termes d'états
- **Diagrammes d'activité** : représentent le comportement d'une opération en termes d'actions



Sources

<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/>

<https://openclassrooms.com/fr/courses/2035826-debutez-lanalyse-logicielle-avec-uml/2035851-uml-c-est-quoi>

<https://openclassrooms.com/fr/courses/2035826-debutez-lanalyse-logicielle-avec-uml/2048781-les-differents-types-de-diagrammes>

<http://www.math-info.univ-paris5.fr/~bouzy/Doc/UML-NotesCours.pdf>

https://fr.wikiversity.org/wiki/Mod%C3%A9lisation_UML/Les_diff%C3%A9rents_types_de_diagramme

[https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique))

https://ufr-segmi.parisnanterre.fr/medias/fichier/cours1_1109021777473.pdf