

# OML Analyse et Conception de Base de Données



Sérigne Modou DIOP  
Daour Samba NDAW  
Mamadou Hamady THIAM

# PLAN

01

## Approche de la décomposition fonctionnelle: Méthode MERISE

Demarche   Avantages   Inconvénients

02

## Approche Orientée Objet: UML

Demarche   Avantages   Inconvénients

03

## UML: Unified Model Language

Approche 4 vues + 1(Vue des besoins)

13 diagramme UML

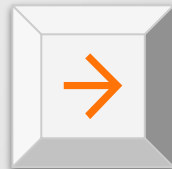
Classification des diagrammes



# Introduction



You can enter a subtitle here if you need it

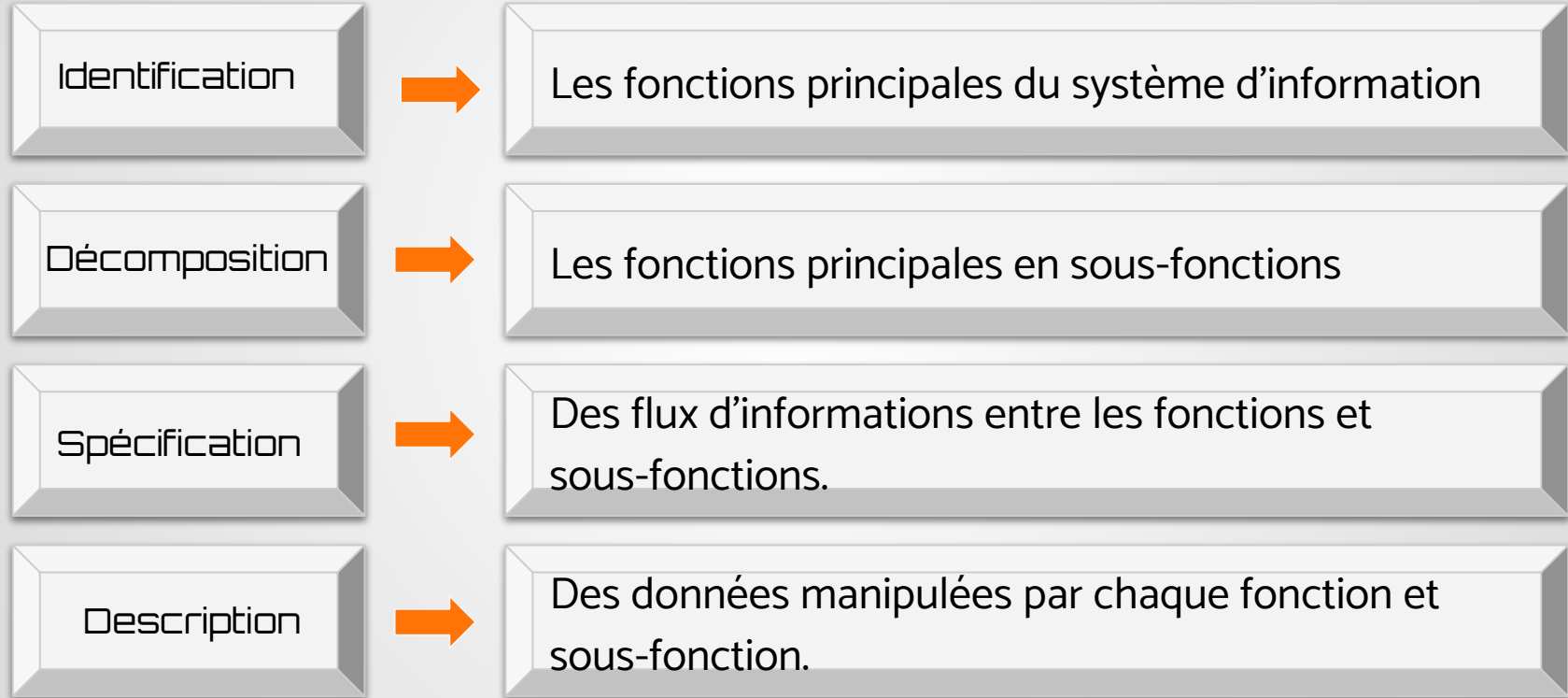


# Approche de la Décomposition fonctionnelle

C'est une méthode d'analyse et de conception des systèmes d'information qui consiste à décomposer un système complexe en sous-systèmes plus simples et cohérents.



## Demarche





## Avantage



Permet une analyse systématique et rigoureuse des fonctions et des processus



Modélisation des fonctions et des données dans des diagrammes spécifiques



Facilite la communication entre les différentes parties prenantes impliquées dans la conception d'un système d'information



Facilite la réutilisation de fonctions et de sous-systèmes éprouvés dans d'autres systèmes d'information



## Inconvénients



Peut être très détaillée et complexe, ce qui peut rendre la conception et la documentation laborieuses et fastidieuses.



Peut conduire à une sur-fragmentation du système d'information, où les fonctions et les sous-fonctions sont décomposées à un niveau excessif de détail



Peut ne pas être adaptée aux systèmes d'information complexes ou aux systèmes en constante évolution

# Approche Orienté Objet: UML



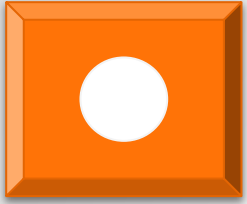
UML

Unified Modeling Language (Langage de Modélisation Unifié)

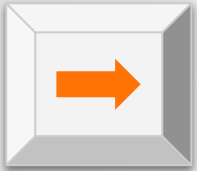


Permet de modéliser les systèmes d'information en termes d'objets et de relations entre ces objets.

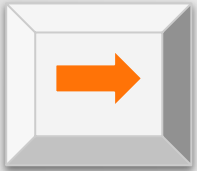




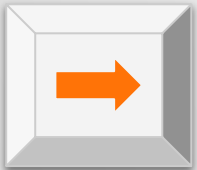
# Démarche



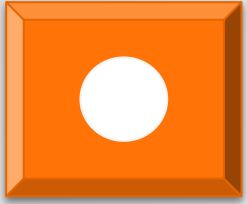
Analyse



Conception



Realisation



# Avantages



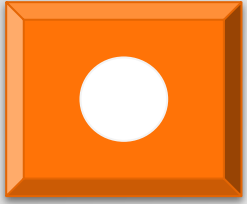
Comprendre les besoins de l'entreprise et de formaliser ces besoins en termes d'objets.



Bien organiser les données et de mettre en place un système d'information performant et facilement maintenable.



Plus souple que la méthode Merise et permet de répondre à des besoins évolutifs.



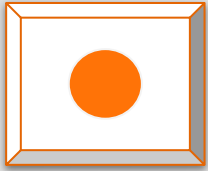
# Inconvénients



La méthode UML est également assez complexe à mettre en œuvre



Nécessite une bonne connaissance des processus métier de l'entreprise.



# Etudes comparative entre Merise et UML



UML est plus souple et permet de répondre à des besoins évolutifs.



UML est également mieux adapté pour la modélisation d'applications complexes et pour la conception d'applications orientées objet.



En revanche, la méthode Merise est plus adaptée pour la modélisation de bases de données relationnelles simples.

# **UML** Unified Modeling Language

**UML est une méthode de modélisation orientée objet utilisée pour concevoir des systèmes logiciels.**

**Elle se concentre sur la modélisation du comportement et de la structure des systèmes logiciels.**

**Elle est adaptée aux projets de grande envergure, complexes et impliquant plusieurs acteurs.**

# Approche Vues

En UML, il existe plusieurs vues pour modéliser différents aspects d'un système. Voici une brève explication de ces vues :

## **La vue logique**

modélise les concepts métiers et les relations entre eux. Elle se concentre sur la représentation des classes, des objets, des associations, des héritages et des relations entre les objets.

## **La vue processus**

modélise les processus ou les activités impliquées dans le système. Elle se concentre sur la représentation des diagrammes d'activité, des états et des transitions.

## **La vue composant**

modélise les composants physiques du système. Elle se concentre sur la représentation des composants, de leurs interfaces et des dépendances entre eux.

# Approche Vues (suite)

## **La vue de déploiement :**

elle modélise la façon dont le système est déployé sur les machines physiques. Elle se concentre sur la représentation des nœuds, des connexions réseau, des composants déployés et de leurs relations.

En plus de ces vues, il y a aussi une vue des besoins qui modélise **les exigences et les besoins des utilisateurs**. Cette vue se concentre sur la représentation des cas d'utilisation, des acteurs, des scénarios d'utilisation et des exigences fonctionnelles et non fonctionnelles du système.

# Les 13 Diagrammes UML

- |                                   |                                      |
|-----------------------------------|--------------------------------------|
| 1- Diagramme de classes           | 8- Diagramme de composants           |
| 2- Diagramme d'objets             | 9- Diagramme de déploiement          |
| 3- Diagramme de cas d'utilisation | 10- Diagramme de packages            |
| 4- Diagramme de séquence          | 11- Diagramme de profils             |
| 5- Diagramme de collaboration     | 12- Diagramme de structure composite |
| 6- Diagramme d'états              | 13- Diagramme de timing              |
| 7- Diagramme d'activités          |                                      |



# Classification des diagrammes

Classification des diagrammes selon l'aspect architecture :

- Diagramme de classes
- Diagramme d'objets
- Diagramme de composants
- Diagramme de déploiement

Classification des diagrammes selon l'aspect fonctionnel :

- Diagramme de cas d'utilisation
- Diagramme d'activité
- Diagramme de séquence
- Diagramme de communication
- Diagramme d'états-transitions

Chaque type de diagramme UML a sa propre utilité et permet de représenter une partie spécifique d'un système. En combinant ces diagrammes, il est possible de créer une représentation complète et cohérente du système.

# Thank you!

