

LES DIAGRAMMES DANS LA CONCEPTION DE PROJET INFORMATIQUE

SOMMAIRE

I. GANTT

II. UML

- A) Diagramme de cas d'utilisation
 - B) Diagramme de classe
 - C) Diagramme de séquence
 - D) Diagramme d'activités
-
- III. MERISE
 - A) Modèle Conceptuel de Données (MCD)
 - B) Modèle Logique de Données (MLD)
 - C) Modèle Physique de Données (MPD)

IV. PERT

GANTT

- Outil de planification
- Représentation visuelle des tâches d'un projet sur une durée.
- Objectifs:
 - décomposer le projet en phase (Idéation /Conception / Design / Production...)
 - Suivre l'avancement des tâches
 - Visualiser les échéances et dépendances
 -
- Les outils :
 - Trello
 - Confluence / Jira
 - Notion
 - Excel / Google sheets
 - The Gantt Project

UML (Unified Modeling Language)

- Langage de modélisation standardisé pour représenter les différents aspects d'un système informatique
- On retrouve **plusieurs types de diagrammes** :
 - **Cas d'utilisation** : montrer les interactions entre users et system --> illustrations des fonctionnalités principales
 - **Classe** : représentation de la structure des classes d'un système + leurs relations
 - **Séquence** : représentation de l'interaction des objets dans le temps
 - **Diagramme d'activités** : visualiser les flux d'activités et les processus

CAS D'UTILISATION (UC)

- Description des **fonctions générales** et portée d'un système.
- Identification les interactions entre le système et les acteurs et les limites (qu'est-ce que fait le système et comment les acteurs l'utilisent).
- Vision utilisateur et pas technique / informatique
- Contexte :
 - réalisation du diagramme au début du projet
 - phase de tests
- Les éléments du diagrammes sont :
 - les acteurs
 - les cas d'utilisations (use case)
 - les relations d'associations

CAS D'UTILISATION (UC)

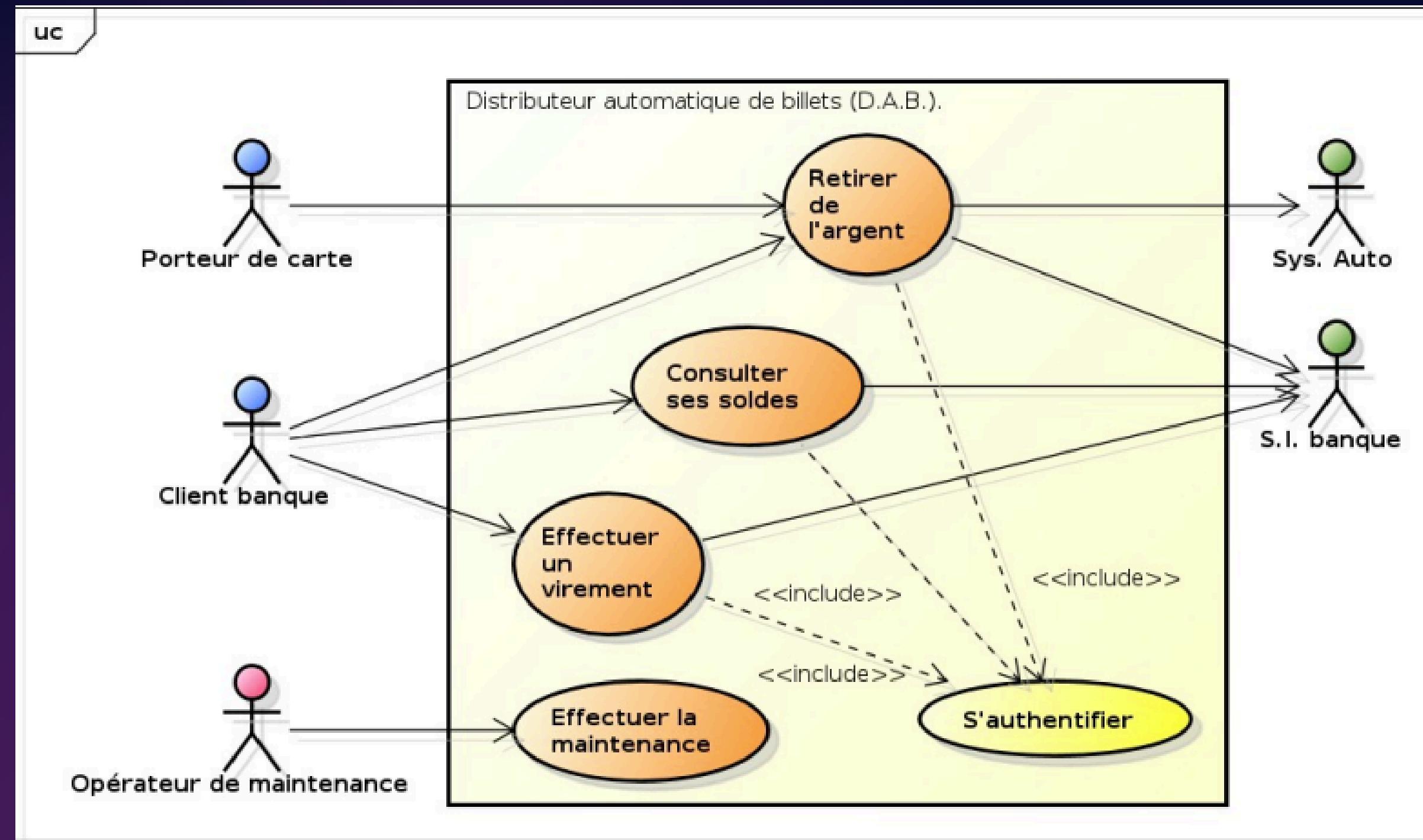


DIAGRAMME DE CLASSE

- Classe : objet ou ensemble d'objets partageant une structure et un comportement communs.
- Modélisation de la structure statique d'un package ou d'un système complet
- Modélisation des objets composant le système et représentation des relations entre les différents objets
- Comprendre les exigences du système et identifier ses composants.
- Les éléments :
 - les classes (représentées par un rectangle divisés en trois sections nom / attributs et méthodes)
 - Relations entre classes
 - association
 - héritage (sous-classe héritant des propriétés de la classe parent)
 - dépendance (indiquée par des flèches en pointillées)

DIAGRAMME DE CLASSE

- Les symboles 0..1, 1,0,0*,1..* etc représentent la multiplicité des relations entre les classes
- **1** = il doit y avoir **exactement une instance de la classe associée**.
 - Ex : si j'ai une classe "**Personne**" qui est liée à une classe "**Passeport**" avec une multiplicité de 1 du côté de "**Passeport**" alors chaque "**Personne**" doit posséder exactement un passeport .
- **0..1** = il peut y avoir zéro ou une seule instance de la classe associée
 - Ex : Une classe "**Personne**" liée à une classe "**Voiture**" à une muliticiplicité de 0..1 du côté de "**Voiture**"
- **0..*** ou **0..n** = Une instance peut être associée à aucune, une ou plusieurs instances de la seconde classe.
 - Ex : "**Bibliothèque**"--> "**Livres**" avec 0..* à côté de "**Bibliothèque**" signifie que ma classe "Bibliothèque" peut posséder aucun, un ou plusieurs livres

DIAGRAMME DE CLASSE

Exemple de la bibliothèque (5)

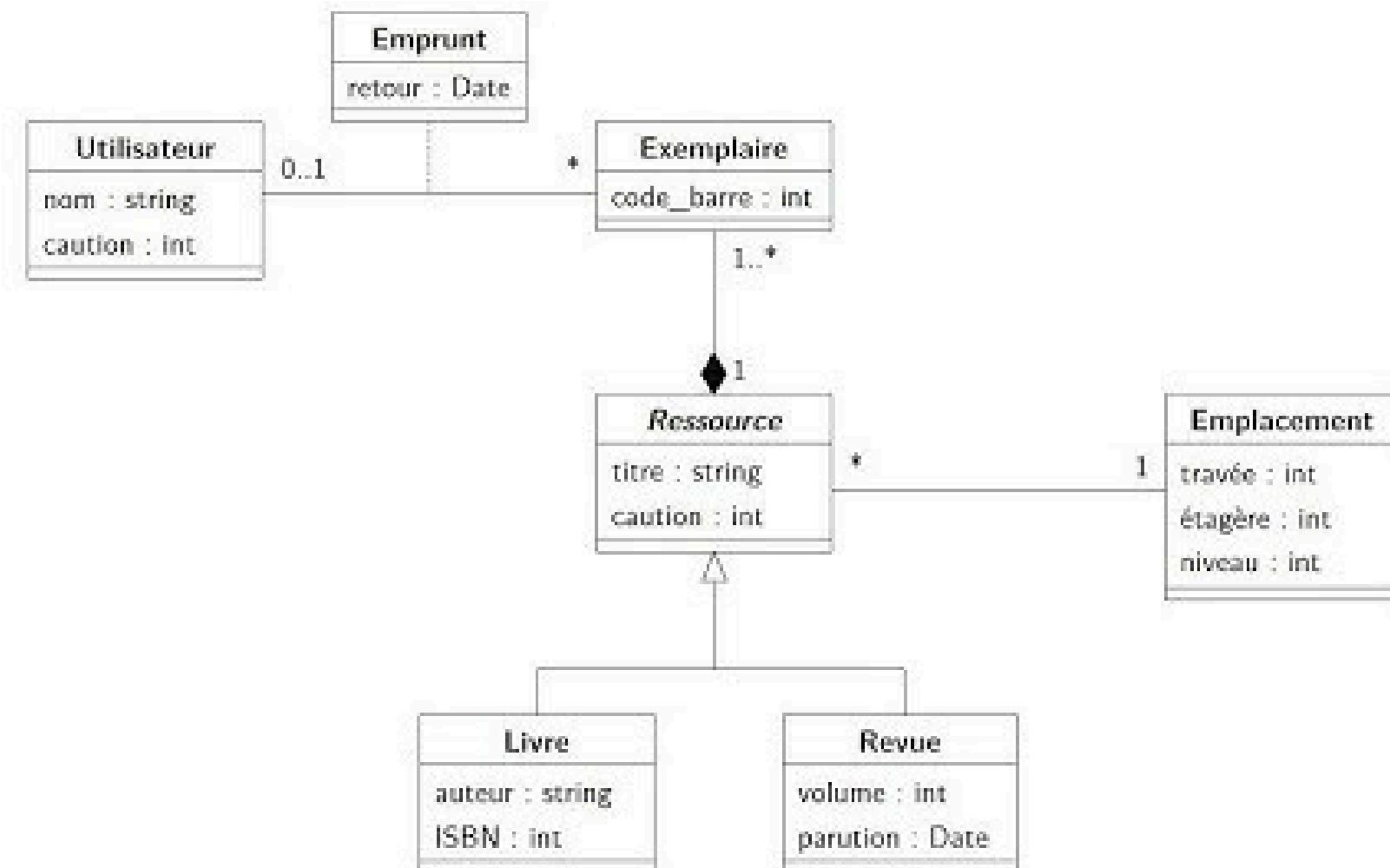


DIAGRAMME DE SÉQUENCE

- Représente l'interaction d'un ensemble d'objets dans un processus *au fil du temps* .
- Affichage des messages qui passent entre les acteurs et objets dans le système ainsi que l'ordre dans lequel ils se produisent.
- Les éléments du diagramme :
 - Objet : représente les objets utilisés. Chaque objet est représenté par un carré surmontant une ligne en pointillé. Cette ligne représente la durée de vie de l'objet.
 - Période d'activation d'un objet : sur la ligne, on insère des périodes d'actions d'un objet --> représenter le moment où un objet est actif
 - Message : représenté par des flèches horizontales --> messages échangés entre les différents objets. L'ordre d'envoi est indiqué par la position des flèches sur l'axe vertical.
 - Paquetage: divise et organise la représentation du diagramme (un peu comme des dossiers organisent des fichiers).

DIAGRAMME DE SÉQUENCE

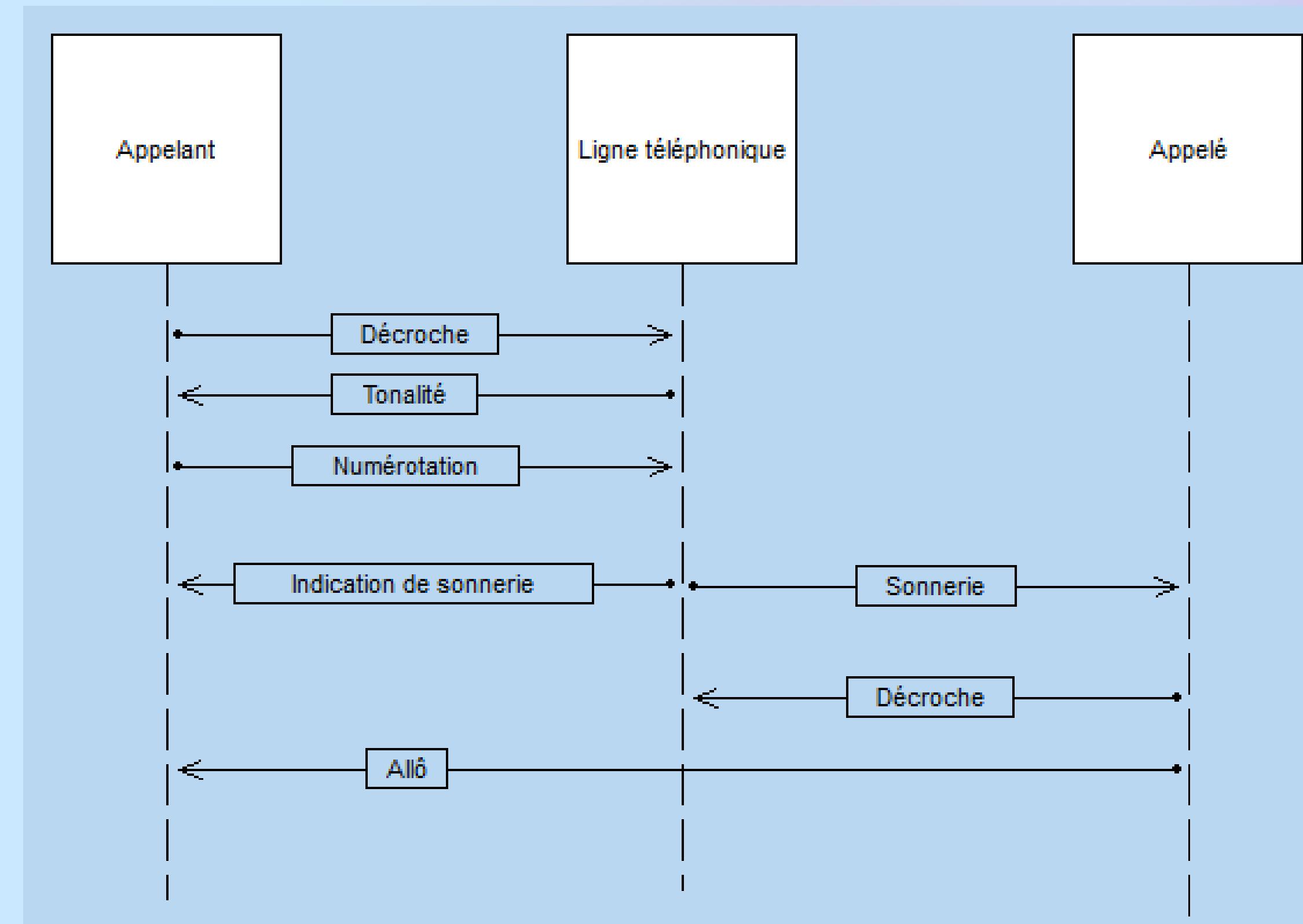


DIAGRAMME D'ACTIVITÉS

- Fournit une vue du comportement d'un système décrivant la séquence d'actions d'un processus.
- Similaire aux organigrammes de traitement de l'information.
- Contexte:
 - avant le lancement d'un projet --> modéliser les principaux flux de travaux.
 - durant la phase d'analyse et de conception --> définir les comportements des opérations

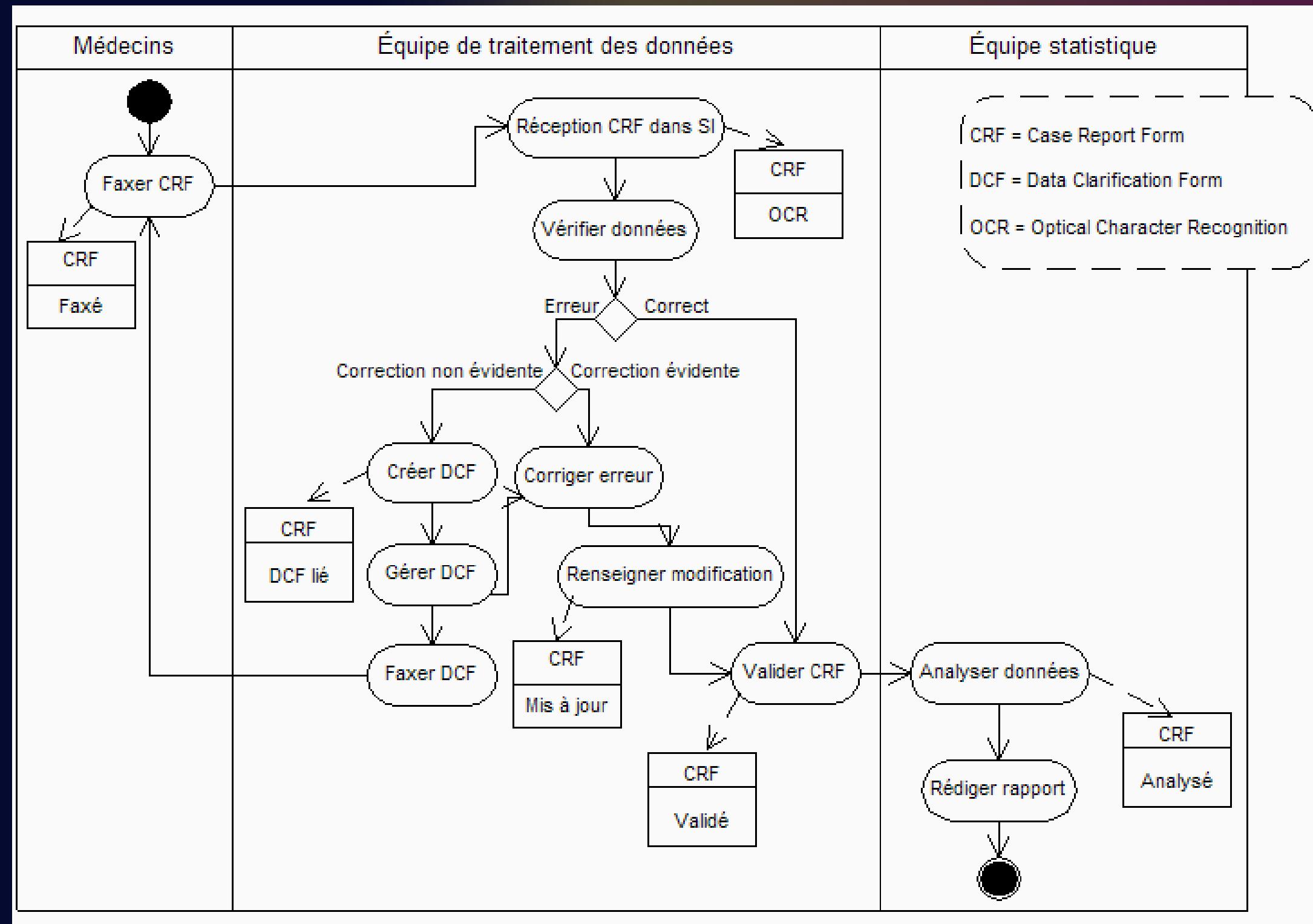
DIAGRAMME D'ACTIVITÉS

- Les éléments du diagramme d'activité :
 - Activités (ou Actions) :
 - représentées par des rectangles avec des coins arrondis
 - représente une tâche ou une étape du processus
 - Flux de contrôle :
 - représentés par des flèches qui relient les activités entre elles
 - montrent l'ordre d'exécutions des tâches
 - Début et fin :
 - début : représenté par un cercle rempli indiquant le début du process
 - fin : représenté par un cercle avec un contour épais
 - Noeuds de décision :
 - représentés par des losanges --> modélisent les choix ou conditions dans le flux
(user logged ? yes/no)

DIAGRAMME D'ACTIVITÉS

- Fourches et jonctions :
 - Fourches : trait épais horizontal ou vertical séparant le flux en branches parallèles
 - Jonctions : Même symbole utilisé pour synchroniser plusieurs flux parallèles en un seul
- Swimlanes (ou couloirs)
 - Représentent les rôles ou acteurs qui effectuent les actions
- Objets (optionnel) :
 - peuvent être placés sur le diagramme pour montrer les données utilisées lors des actions
 - Flux d'objets:
 - permettent de visualiser le passage d'objet ou de données entre les actions

DIAGRAMME D'ACTIVITÉS



III. MÉTHODE MERISE

- Méthode française de conception de système d'information
- Propose une approche structurée pour modéliser les données et traitement
- Plusieurs diagramme:
 - Modèle conceptuel de données (MCD) : décrit les données à gérer dans le système (entités attributs relations)
 - Modèle Logique de données (MLD) : traduit le MCD en une forme plus proche du modèle de la DB (relationnel / hiérarchique)
 - Modèle Physique de données (MPD) : s'appuie sur le MLD pour décrire la structure finale de la base de données
- Contexte : utilisé pour les projets S.I en entreprise

DIAGRAMME MCD

MODÈLE CONCEPTUEL DE DONNÉES

- Représentation schématique illustrant les données d'un système.
- Sert à organiser, structurer et visualiser ces données de façon logique et compréhensible
- Se concentre sur l'aspect conceptuel. (on ne retrouve des détails techniques par exemple)
- Objectif --> comprendre comment les données sont connectées et interagissent entre elles.

DIAGRAMME MCD

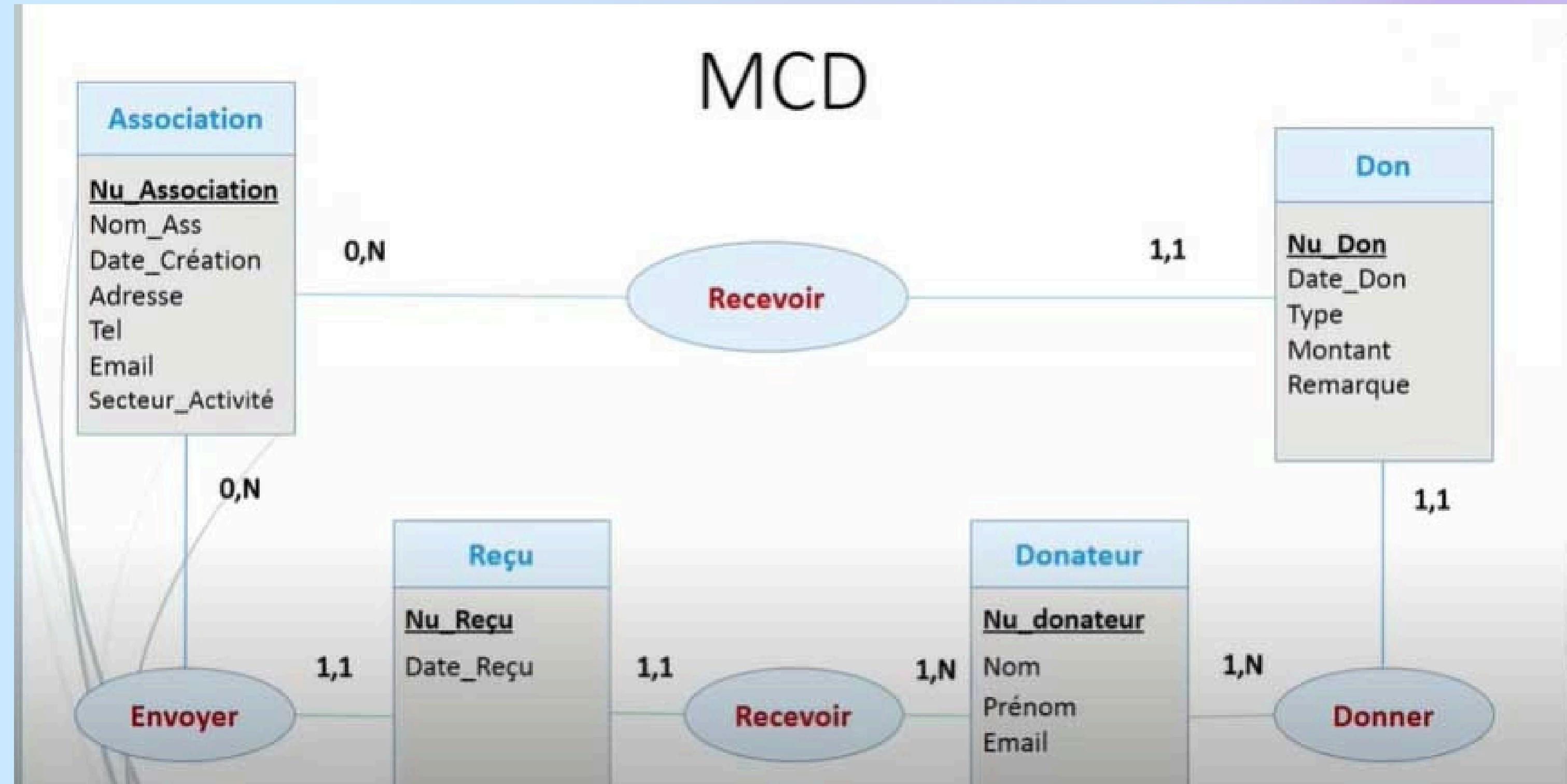


DIAGRAMME MLD

MODÈLE LOGIQUE DE DONNÉES

- Organiser les données sous forme compatible avec les bases de données relationnelles
- Eviter les redondances et anomalies
- Préparer le MPD (ultime étape pour implémenter la DB)
- Le MLD est composé de tables et relations:
 - Tables : (issues du MCD). Chaque table représente une entité du système. Ex : "client", "commandes" ...
 - Colonnes : attributs de l'entité. Ex : "nom", "adresse" pour "Client".
 - Clé primaire (PK) : un ou plusieurs attributs permettant d'identifier chaque enregistrement d'une table
 - Clés étrangères (FK) : représente les relations entre les tables. Chaque table en possède une.

DIAGRAMME MLD

MODÈLE LOGIQUE DE DONNÉES

- Relations
 - 1:1 (un à un): Par exemple, chaque Personne est associée à un “Passeport”.
 - 1 (un à plusieurs) : Par exemple, un Client peut passer plusieurs Commandes.
 - N : (plusieurs à plusieurs) : Par exemple, une Commande peut contenir plusieurs Produits, et chaque produit peut apparaître dans plusieurs Commandes.

DIAGRAMME MPD

MODÈLE PHYSIQUE DE DONNÉES

- Dernière étape de la méthode Merise pour la conception d'une db.
- Objectif --> transformer le MLD en une représentation plus concrète pour être implémenté dans un système de gestion de DB.
- Ce diagramme détaille toutes les spé nécessaires pour créer et structurer les tables, colonnes, types de données...)

RÉCAPITULATIF

Diagramme de Classes	UML	Décrit la structure statique d'un système orienté objet avec les classes, leurs attributs, méthodes, et relations.	Conception et modélisation des structures logiques du système.
Diagramme de Cas d'Utilisation	UML	Montre les interactions entre le système et les acteurs pour illustrer les fonctionnalités principales.	Définition des fonctionnalités attendues du système.
Diagramme de Séquence	UML	Représente l'ordre des interactions entre objets dans le temps, pour une fonctionnalité ou un scénario spécifique.	Modélisation dynamique des flux d'information.
Diagramme d'Activités	UML	Modélise le flux d'activités ou de processus, avec des décisions et des parallélismes éventuels.	Représentation des processus métiers ou des flux de travail.

RÉCAPITULATIF

Diagramme	Méthode	Objectif principal	Utilisation principale
Modèle Conceptuel de Données (MCD)	Merise	Représente la structure des données de façon indépendante du système, identifiant les entités, leurs attributs, et relations.	Modélisation conceptuelle des données, analyse métier.
Modèle Logique de Données (MLD)	Merise	Transforme le MCD en une structure adaptée aux bases de données relationnelles, avec des tables et des clés étrangères.	Préparation de la base de données en lien avec le MCD.
Modèle Physique de Données (MPD)	Merise	Décrit la structure détaillée de la base de données, incluant les types de données et les spécifications physiques.	Conception physique de la base de données pour l'implémentation.