

Module  
Développement Orienté Objet Avancé

# UML : Analyse et conception OO II

---

Pr. Zineb MCHARFI

# Diagramme de communication

Répond à la question « **Comment** »

**Utilisé par** : BA, Architecte, Développeur.

- Diagramme dynamique (**interactions**).
- Modélise les interactions entre des objets pour réaliser une action.
- Créé en phase de conception.
- Même principe que le diagramme de séquence, mais avec une représentation spatiale et non temporelle.
- Appelé également diagramme de collaboration (anciennes versions UML).
- Chaque action du diagramme d'activités peut être détaillée dans un diagramme de communication

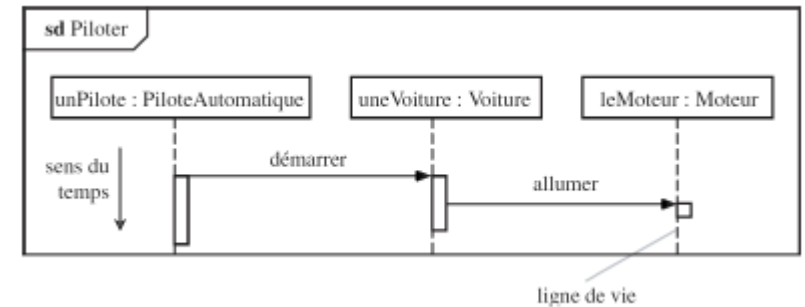


Diagramme de séquence

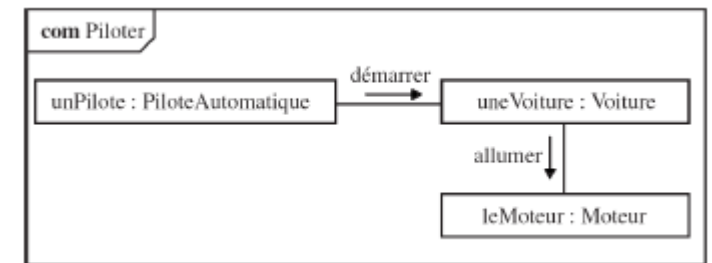
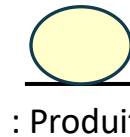
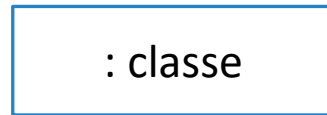


Diagramme de communication

# Diagramme de communication - Concepts

- Représentation des **objets** : idem que pour diagramme de séquence.

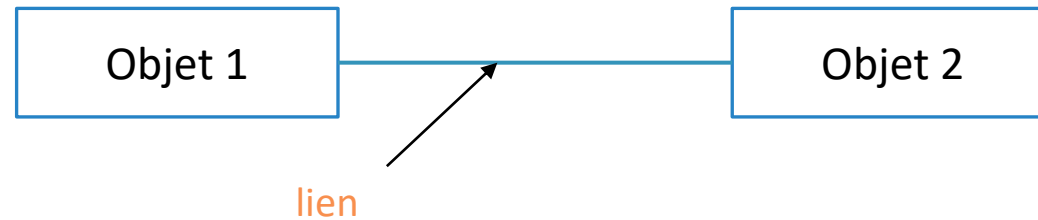


Etiquette: [objet]:[classe]

# Diagramme de communication - Concepts

---

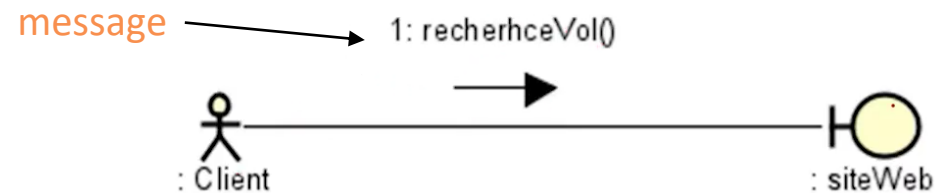
- **Lien (connecteur)** : connexion entre deux objets.


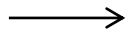
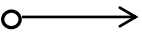


# Diagramme de communication - Concepts

## ■ Message :

- Déclenche l'exécution d'une opération de l'objet destinataire.
- On peut avoir plusieurs messages sur un même lien.



- **Types de messages** : synchrone:  asynchrone:  retour: 
- **Syntaxe** :  
[prédécesseurs/][garde]**[séquence]**[itération][valeurRetour:=]**signatureMessage**(arguments)

# Diagramme de communication - Concepts

---

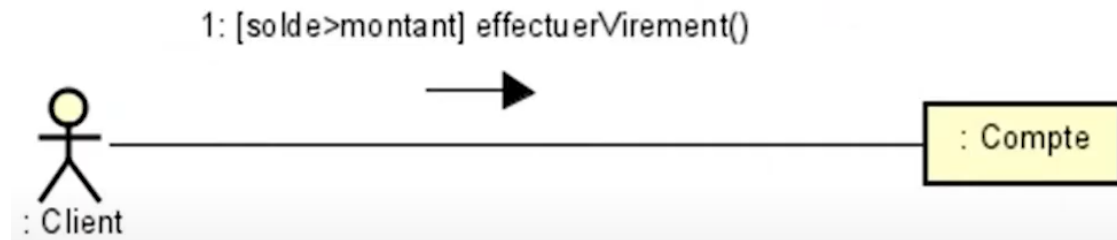
## ■ Message – Séquence (numérotation) :

- Remplace l'aspect temporel du diagramme de séquence -> ordre d'exécution des messages.
- Numérotation simple: 1, 2, 3, ....
- Séquencement : 1, 1.1, 1.2, 1.3 : Le point 1 ne peut se terminer que si ses sous-points 1.1, 1.2, 1.3 se terminent.
- Concurrence (parallélisme): 1, 1.1a, 1.1b, 1.2, 1.3: les points 1.1a et 1.1b s'exécutent en //

# Diagramme de communication - Concepts

## ■ Message – gardes :

- Condition d'envoi du message : [condition]



# Diagramme de communication - Concepts

## ■ Message – itérations :

- Envoi du message autant de fois que le nombre d'itérations : [\* itération]





# Diagramme de communication - Concepts

## ■ Message – arguments :

- Paramètres de la méthode.



# Diagramme de communication - Concepts

## ■ Message – Résultat :

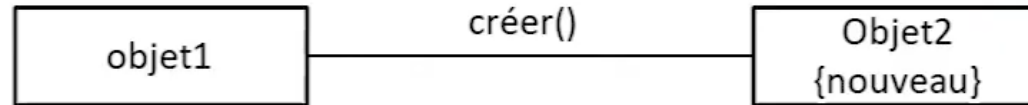
- Valeur retournée par le message.



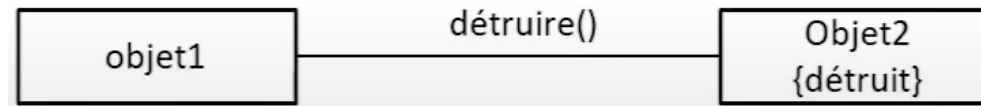
# Diagramme de communication - Concepts

## ■ Message – créer et détruire:

- On ajoute une contrainte associée.
- créer() : ajouter la contrainte {nouveau}



- Détruire() : ajouter la contrainte {détruit}



# Diagramme de communication - Concepts

