

# CONCEPTION ORIENTÉE OBJETS

## UML - Unified Modeling Language

### Travaux pratiques



### Série N° 3

## Diagramme d'objets & Diagramme de Séquence

UML

Youssef EL ALLIOUI

[yelalloui@gmail.com](mailto:yelalloui@gmail.com)

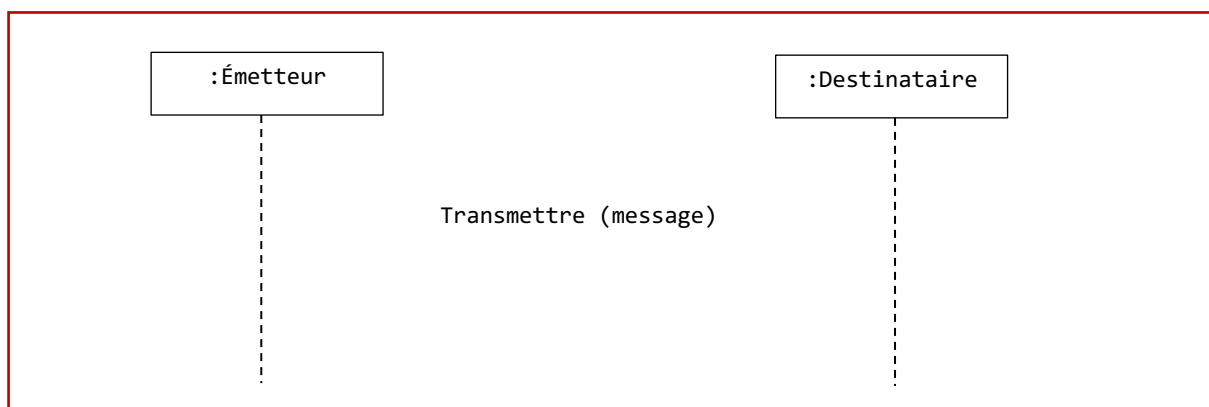
### Exercice 1. Diagramme d'objets

Un objet nommé b747 de classe Avion et en état « détresse » est en relation avec luna, une tour de contrôle. Un ensemble d'autres avions anonymes dont l'état est « à terre » sont aussi liés à luna. La tour de contrôle communique avec p123, une caserne de pompiers.

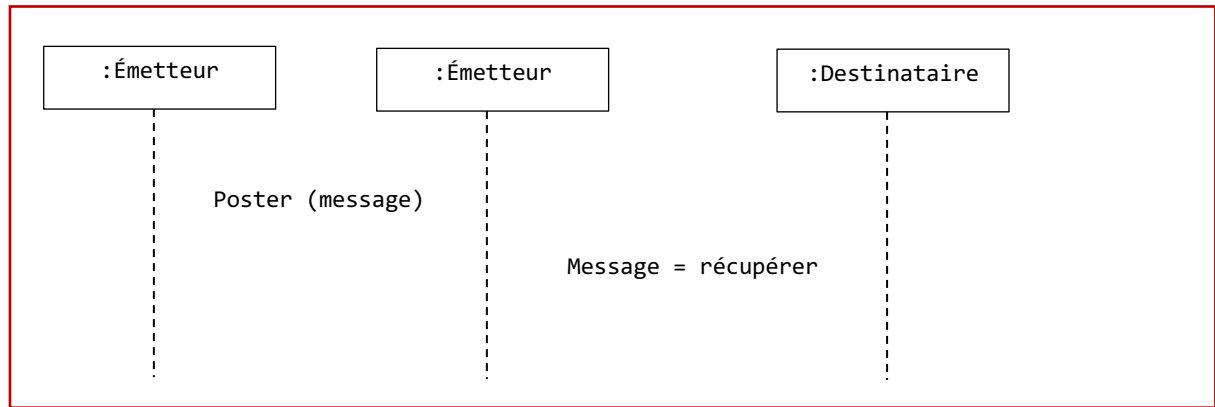
- 1) Dessinez le diagramme d'objets correspondant à la situation décrite ci-dessus.

### Exercice 2. Types de messages

- 1) Quand un courrier électronique est envoyé par l'émetteur, celui-ci ne veut pas attendre que destinataire l'ait reçu et il n'y a pas d'intermédiaire.
  - a. Peut-on utiliser un message synchrone ?
  - b. Complétez la figure ci-dessous par des flèches représentant des messages.

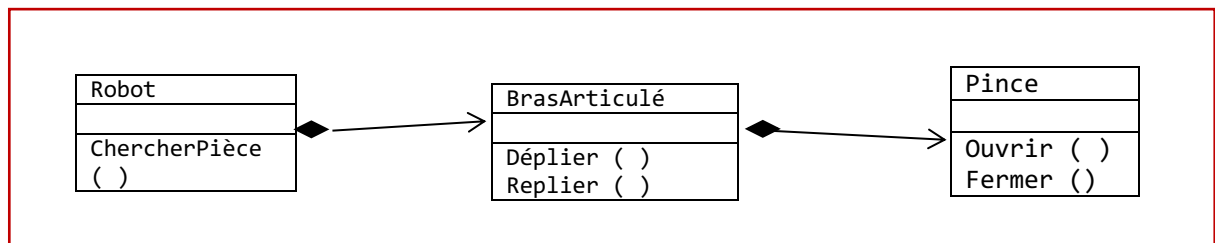


- 2) Est-ce que transmettre est une opération ou un signal ? Dans tous les cas, donnez des éléments d'un diagramme de classe cohérent avec le diagramme de séquence.
- 3) Un serveur de messagerie sert d'intermédiaire entre l'émetteur et le récepteur d'un email. Le serveur est toujours en fonction. Est-ce qu'on peut utiliser des messages synchrones pour l'envoi et la récupération de emails ? Complétez la figure ci-dessous par des flèches représentant des messages.



- 4) Est-ce que poster est une opération ou un signal ? Dans tous les cas, proposez un diagramme de classe cohérent avec le diagramme de séquence.

### Exercice 3. Diagramme de séquence et l'interaction entre classes



Le diagramme de classes ci-dessus décrit des robots composés de bras articulés, eux-mêmes composés de pinces. L'implémentation des classes en question est la suivante :

```

class Robot{
privé :
    BrasArticulé brasArticulé ;
public :
    void chercherPièce() {
        brasArticulé.déplier() ;
        brasArticulé.replier() ;
    }
}
class BrasArticulé {
privé :
    Pince pince ;
public :
    void déplier() {
        pince.fermer() ;
    }
    void replier() {
        pince.ouvrir() ;
    }
}
class Pince {
privé :
    ...
public :
    void fermer() { ... }
}
    
```

```

    void ouvrir() { ... }
}
Début programme principal
    Robot robot ;
    robot.chercherPièce() ;
Fin programme principal

```

- 1) Donnez un diagramme de séquence représentant l'échange de messages résultant de l'exécution du programme principal.

#### Exercice 4. Explication des messages

- 1) Expliquez la syntaxe des messages suivants, extraits d'un diagramme de séquence :

| Message              | Signification |
|----------------------|---------------|
| <b>f</b>             | .....         |
| <b>f (0)</b>         | .....         |
| <b>f (x)</b>         | .....         |
| <b>f (x=0)</b>       | .....         |
| <b>f (y=x)</b>       | .....         |
| <b>f (.)</b>         | .....         |
| <b>f (x, y)</b>      | .....         |
| <b>*</b>             | .....         |
| <b>y = f</b>         | .....         |
| <b>y = f (0)</b>     | .....         |
| <b>y = f (x=0)</b>   | .....         |
| <b>y = f (x) : 0</b> | .....         |