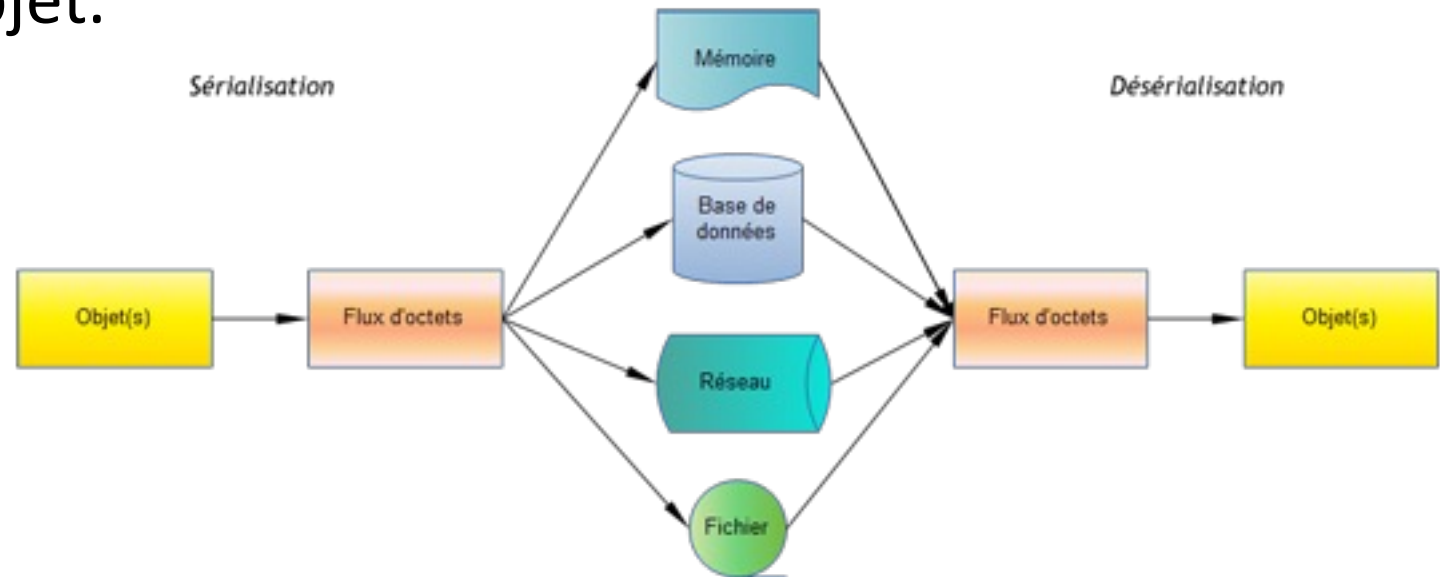


JAVA

La sérialisation

La sérialisation

- La sérialisation, c'est le principe de représenter des données structurées afin de les sauvegarder (dans un fichier ou une base de données) ou des les transmettre (dans un flux réseau par exemple)
- La sérialisation en Java permet de stocker ou transmettre des objets et les attributs de cet objet.
- Permet la **persistance** d'un objet



La sérialisation

- Presque tous les objets Java sont sérialisable, sauf certains éléments du système : Thread, OutputStream et ses sous-classes, Socket, Image, ...
- Pour sérialiser un objet, il doit implémenter l'interface `Serializable`

```
public class MaClasse implements Serializable{  
    private int i;  
    public MaClasse(int i) { }  
    public int getI() { }  
    public void setI() { }  
}
```

Limitations et objets sérialisables

- En général, on ne sérialise que des objets "simples".
- En effet, sérialiser un fichier ou un flux n'a pas de sens, puisque ces objets sont dynamiques et potentiellement *infinis*.
- Sérialiser une référence ou un dictionnaire peut-être ambigu

La sérialisation

- Il n'y a aucune méthode de sérialisation définies sur l'interface `Serializable`
- Pour convertir l'objet en une forme sérialisée, nous pouvons utiliser la méthode `ObjectOutputStream.writeObject()`
- Pour recréer un objet à partir de cette représentation, nous utilisons la méthode `ObjectInputStream.readObject()`
 - Le résultat sera un objet complètement distinct, avec des objets référencés complètement distincts.

La sérialisation : exemple

```
// On crée un flux de sortie vers un fichier
FileOutputStream fichier = new
    FileOutputStream("fichier.txt");
// On crée un flux de sortie d'objet
ObjectOutputStream flux = new
    ObjectOutputStream(fichier);
// On écrit l'objet
flux.writeObject(objet);
```

La désérialisation : exemple

```
// On crée un flux d'entrée vers un fichier
FileInputStream fichier = new
    FileInputStream("fichier.txt");
// On crée un flux d'entrée d'objet
ObjectInput flux = new
    ObjectInputStream(fichier);
// On récupère l'objet
Object objet = (Object) flux.readObject();
```