Serialization and deserialization
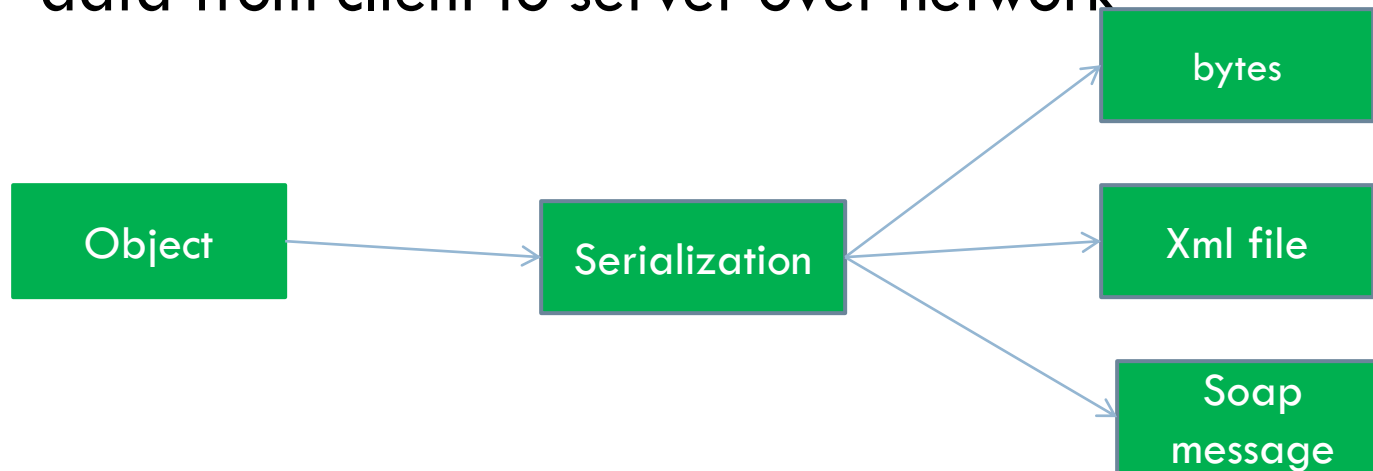
- **Serialization:**
  - Serialization is a process of converting an object into a stream of bytes,xml file, or soap message.
  - Serialization required when transferring or storage data from client to server over network

```
Object ──→ Serialization ──→ bytes
                          ──→ Xml file
                          ──→ Soap message
```
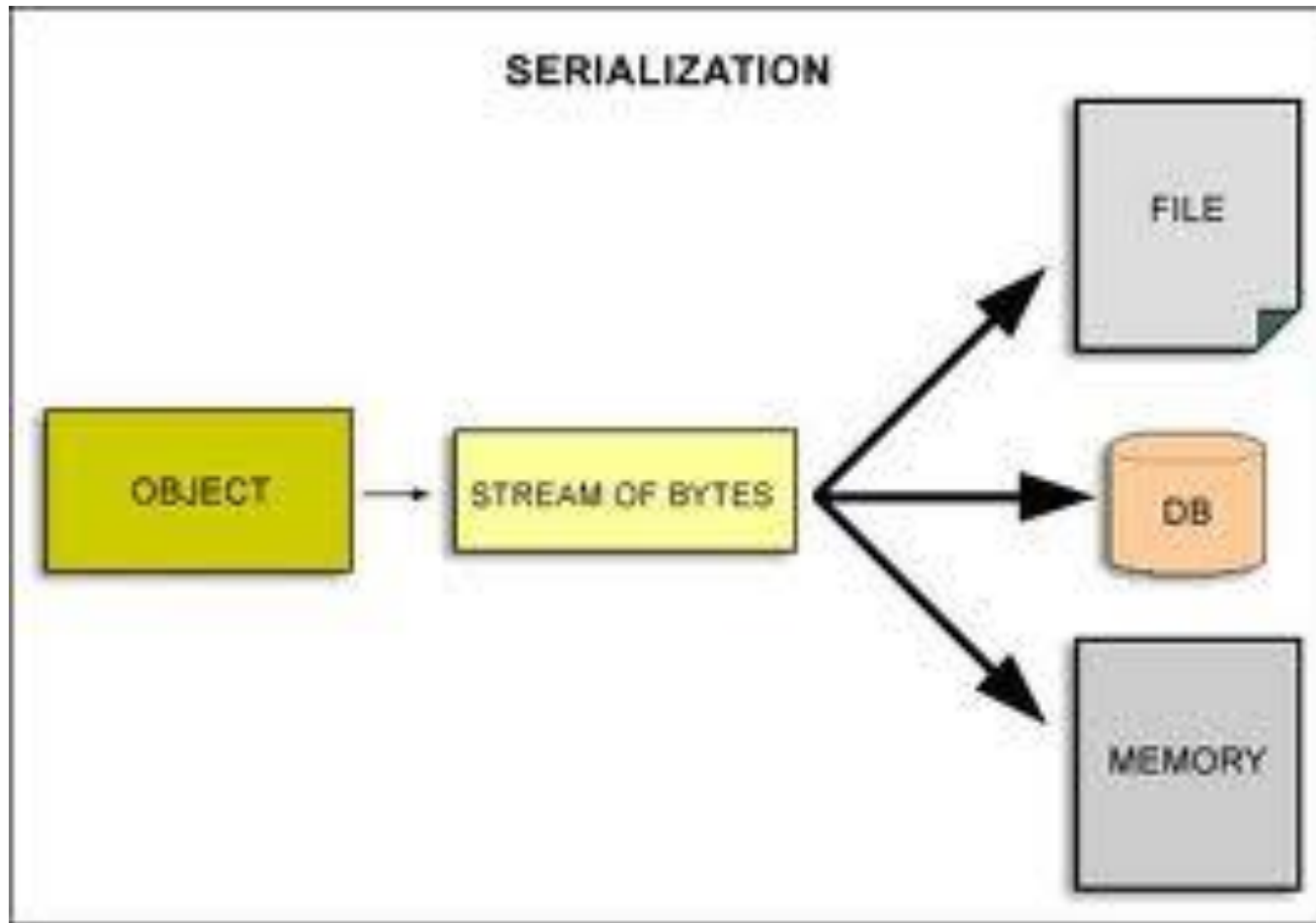
Serialization and deserialization

- **Deserialization:**
    - Deserialization is a process of Reconstruct the object from stream of bytes or xml file or soap message.
- To perform serialization and deserialization they are 3 types of standards.
    - 1.Binary serialization.
    - 2.Xml serialization.
    - 3.Soap serialization
    - 4.JSON serialization
- To work with Serialization .NET comes with a predefined library

    System.Runtime.Serialization
- Advantage of serialization is transportation of an object through a network

Serialization and deserialization

Binary Serialization

- **Binary serialization:**
- In Binary serialization the object is converted into Binary stream.
- Binary serialization used to transport data across the network.
- Binary serialization is known for its light and fast and compact nature
- Remoting uses the Binary Serialization.
- To work with this standard serialization .NET provide a name space

  System.Runtime.Serialization.Formattors.Binary
- Metods of **BinaryFormatter** class
  - void Serialize(Stream serializationStream, object graph)   Convert the object into stream
  - object Deserialize(Stream serializationStream)
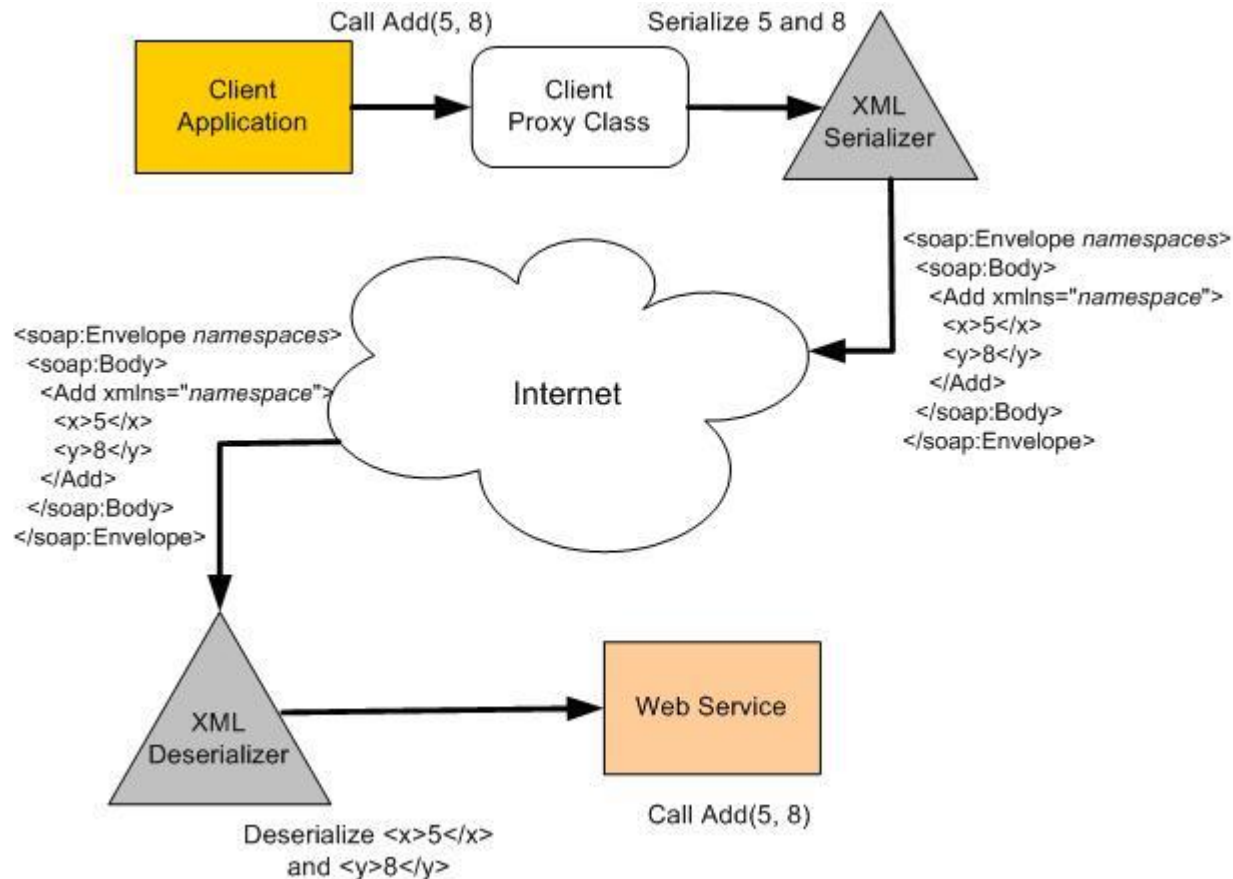   Re construct the object from stream.

Serialization and deserialization

- Serializable attribute: it is an attribute which should mention immediately above the class declaration whose object you want to serialize or deseralize.

[Serializable]

class emp

{

}

Serialization and deserialization

Serialization and deserialization

- xml standard: this is used for maintaining and restoring the object in a xml format
- By using XML serialization only public properties and fields can be serialized.
- XML serialization does not convert methods, indexers, private fields, or read-only properties.
- XML Serialization is much slower compared to Binary serialization.
- To work with this standard .NET provide a name space
- System.Xml.Serialization
- Methods of XmlSerializer :
- void Serialize(Stream **stream, object o);**
- Object Deserialize(Stream **stream);**