

# Persistence

## An Introduction to XML and Serialization

---

Produced by: Dr. Siobhán Drohan  
Mr. Colm Dunphy  
Mr. Diarmuid O'Connor  
Dr. Frank Walsh



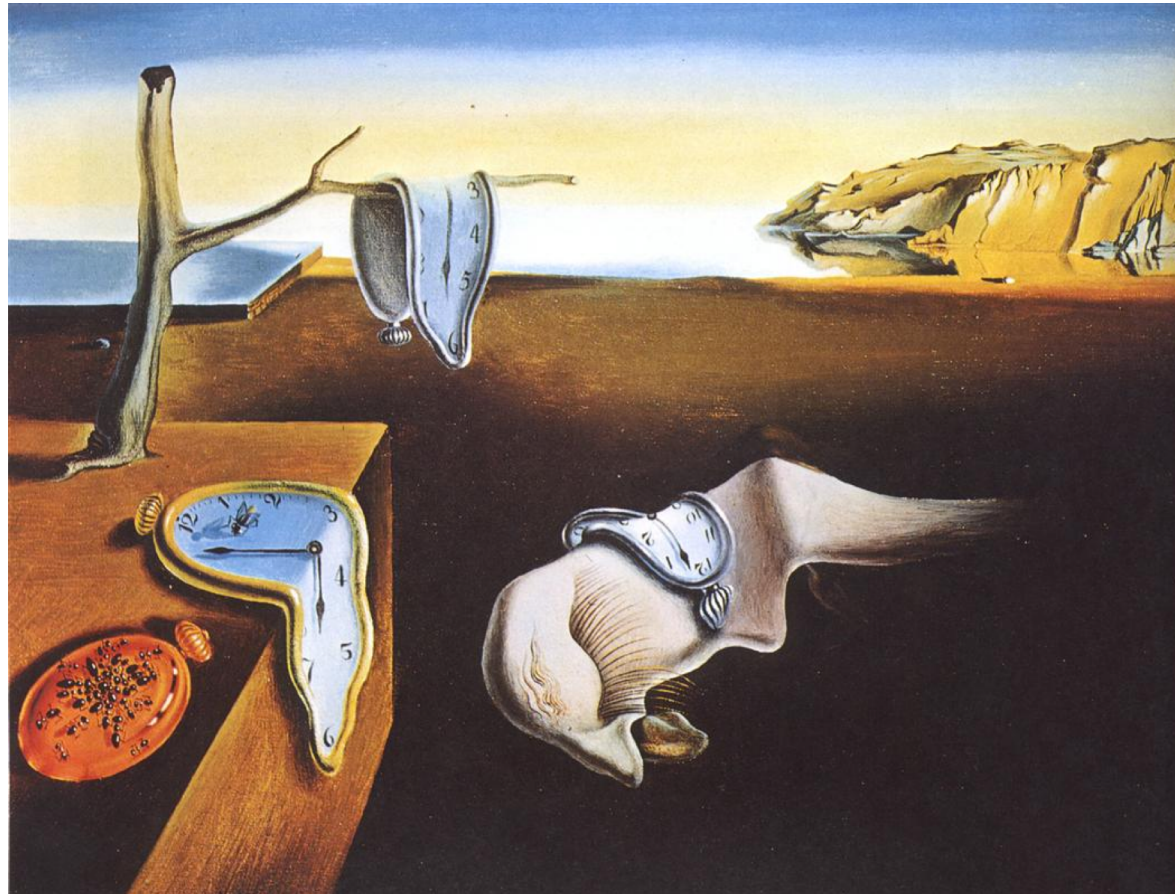
Waterford Institute of Technology  
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics  
<http://www.wit.ie/>

# Persistence

---

“the continued existence of something”



*Dali – “Persistence of memory”*

Persistence – lack of (volatility - volatile)


---



# Persistence - Topic List

---

## 1. Introduction to **XML**:

- 
- XML versus HTML
  - Example of XML
  - XML does not “do” anything

## 2. Object **Serialization**.

# XML versus HTML

---

- **XML** was designed to **describe data**,
  - focus on **what** the data is.
- XML is about carrying information.

```
<note>  
  <to>Donald Duck</to>  
  <from>Minnie Mouse</from>  
  <heading>Reminder</heading>  
  <body>Meeting at 10am today</body>  
</note>
```

# XML versus HTML

---

- **HTML** was designed to **display data**,
  - focus on **how** the data **looks**.
- HTML is about displaying information.

```
<html>

<head>
  <title>My First Web Page</title>
</head>

<body>
  <h1>My First Web Page</h1>
  <p><b>Hello World Wide Web!</b></p>
  <p><i>Hello World Wide Web!</i></p>
  <p><u>Hello World Wide Web!</u></p>
  <p>This is my first web page.</p>
  <p>HTML tags can give <b><i>various</i></b>
  <u>looks and format</u> to the content of this web page.</p>
</body>

</html>
```

# XML is NOT a replacement for HTML

---

```
<note>
  <to>Donald Duck</to>
  <from>Minnie Mouse</from>
  <heading>Reminder</heading>
  <body>Meeting at 10am today</body>
</note>
```

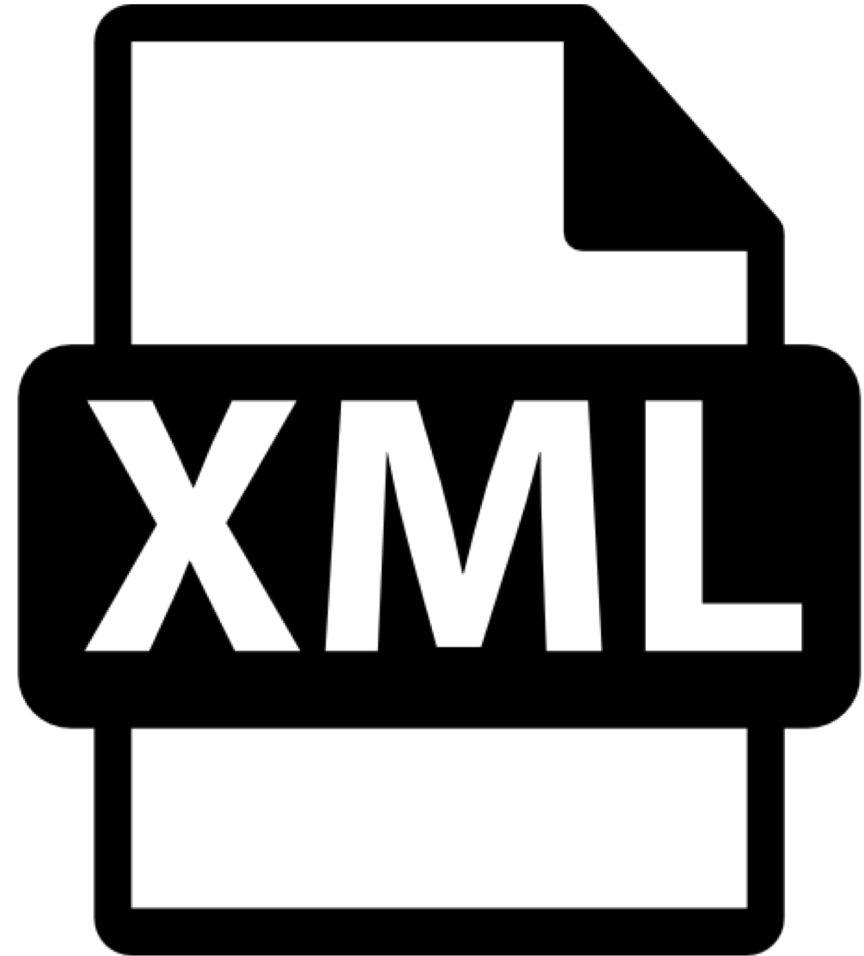


```
<html>
<head>
  <title>My First Web Page</title>
</head>
<body>
  <h1>My First Web Page</h1>
  <p><b>Hello World Wide Web!</b></p>
  <p><i>Hello World Wide Web!</i></p>
  <p><u>Hello World Wide Web!</u></p>
  <p>This is my first web page.</p>
  <p>HTML tags can give <b><i>various</i></b>
  <u>looks and format</u> to the content of this web page.</p>
</body>
</html>
```

# XML...

---

- **Extensible**
  - Extensible markup language
- **Markup**
  - Like HTML
- **Describes Data**
  - Not for displaying data HTML
- **Define your own tags**
  - Not predefined
- **Self Descriptive**





# Persistence - Topic List

---

## 1. Introduction to **XML**:

- XML versus HTML
- Example of XML
- XML does not “do” anything

## 2. Object **Serialization**.

# XML example



```
<note>  
  <to>Donald Duck</to>  
  <from>Minnie Mouse</from>  
  <heading>Reminder</heading>  
  <body>Meeting at 10am today</body>  
</note>
```

A **note** to Donald Duck, from Minnie Mouse, stored as XML

It has sender **<to>** and receiver **<from>** information  
It also has a **heading** and a message **body**.  
The XML is self descriptive.

# Persistence - Topic List

---

## 1. Introduction to **XML**:

- XML versus HTML
- Example of XML
- XML does not “do” anything

## 2. Object **Serialization**.

# Our Shop App



## Shop V4.0

implemented the **CRUD** process



Problem: All entered **data is lost** if we **close our application** (or lose power)

## Shop V5.0

use XML to make our **data persistent** beyond the life of our app



Solution: **Store our objects from memory to XML files.**

# XML does not “do” anything

---

- XML is just information wrapped in <tags>.
- Someone must write a piece of software to send, receive or display it.
- We will write Java code to:
  - **SEND objects TO** an **XML file** on the hard disk.
  - **READ objects FROM** an **XML file** on the hard disk.
- This is called **Object Serialization**

# Persistence - Topic List

---

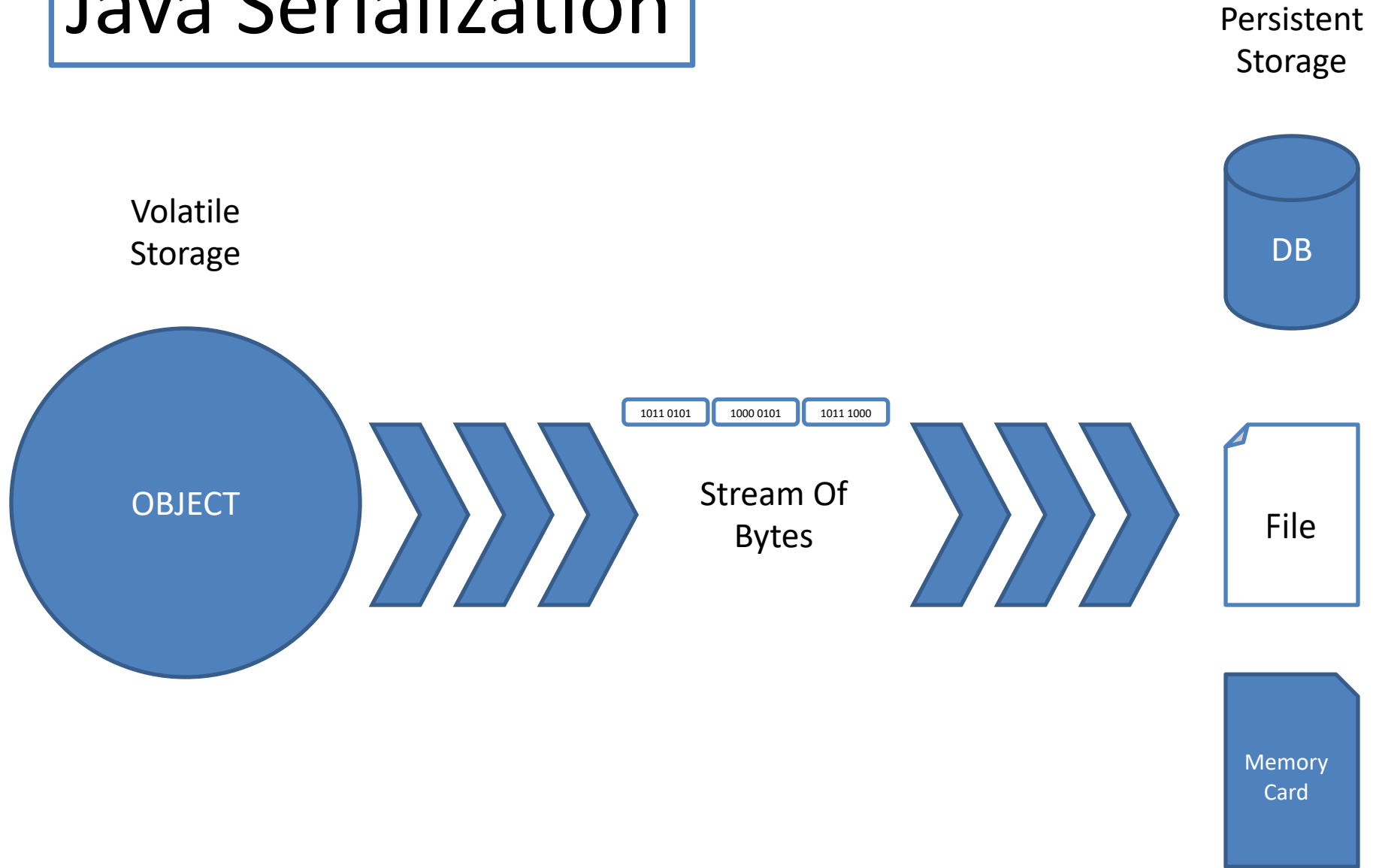
## 1. Introduction to **XML**:

- XML versus HTML
- Example of XML
- XML does not “do” anything

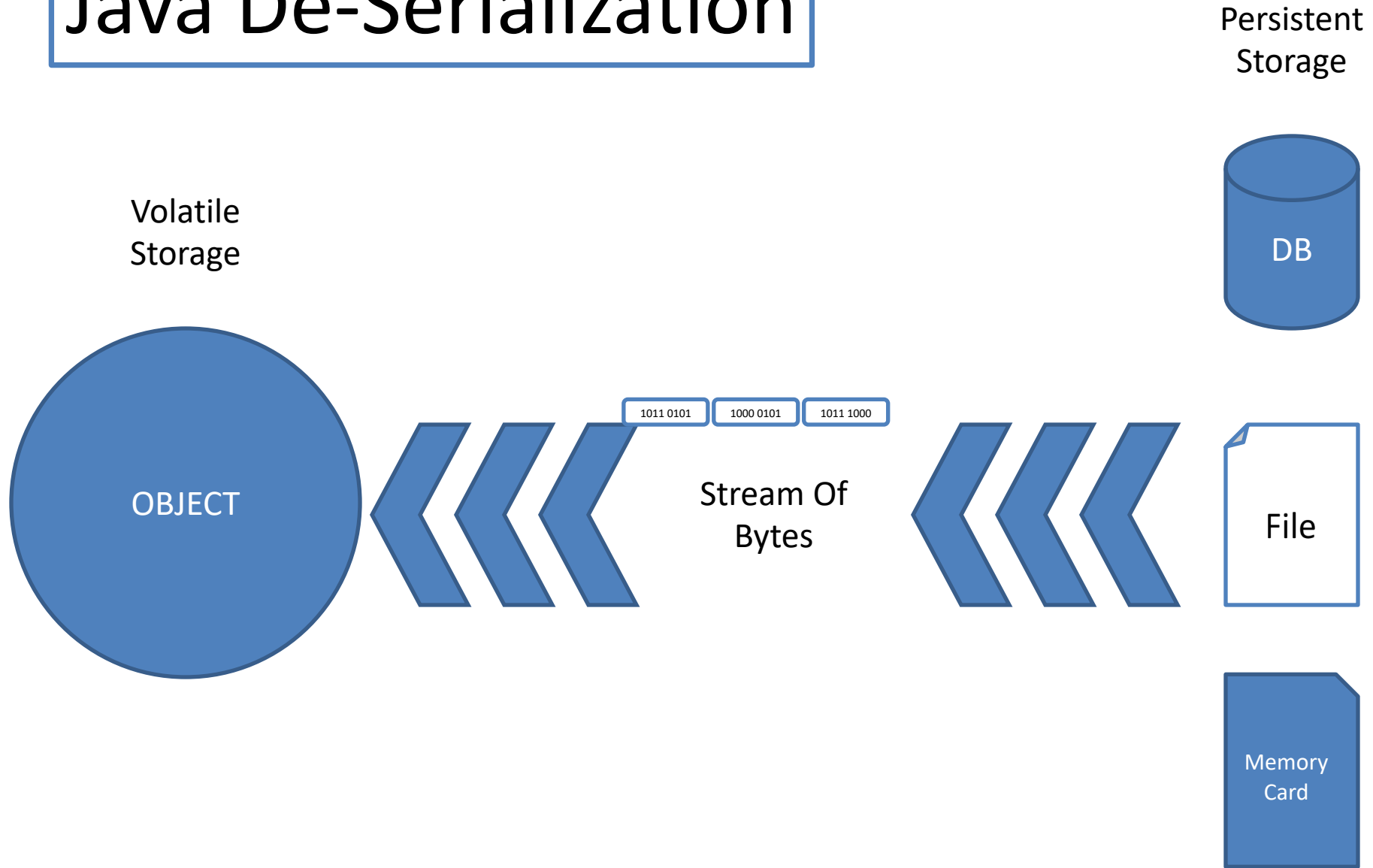


## 2. Object **Serialization**.

# Java Serialization

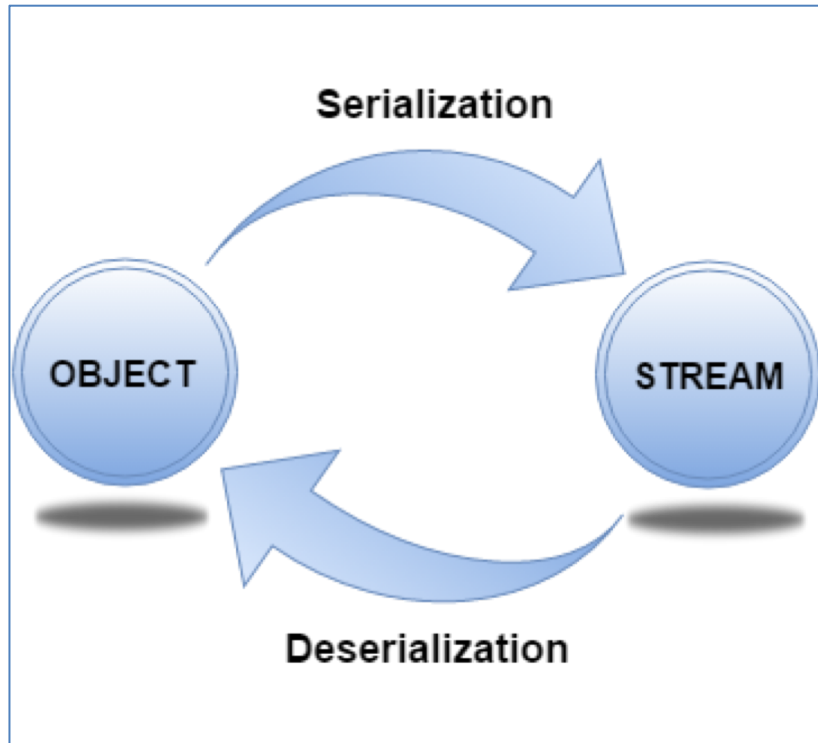


# Java De-Serialization





# Object Serialization

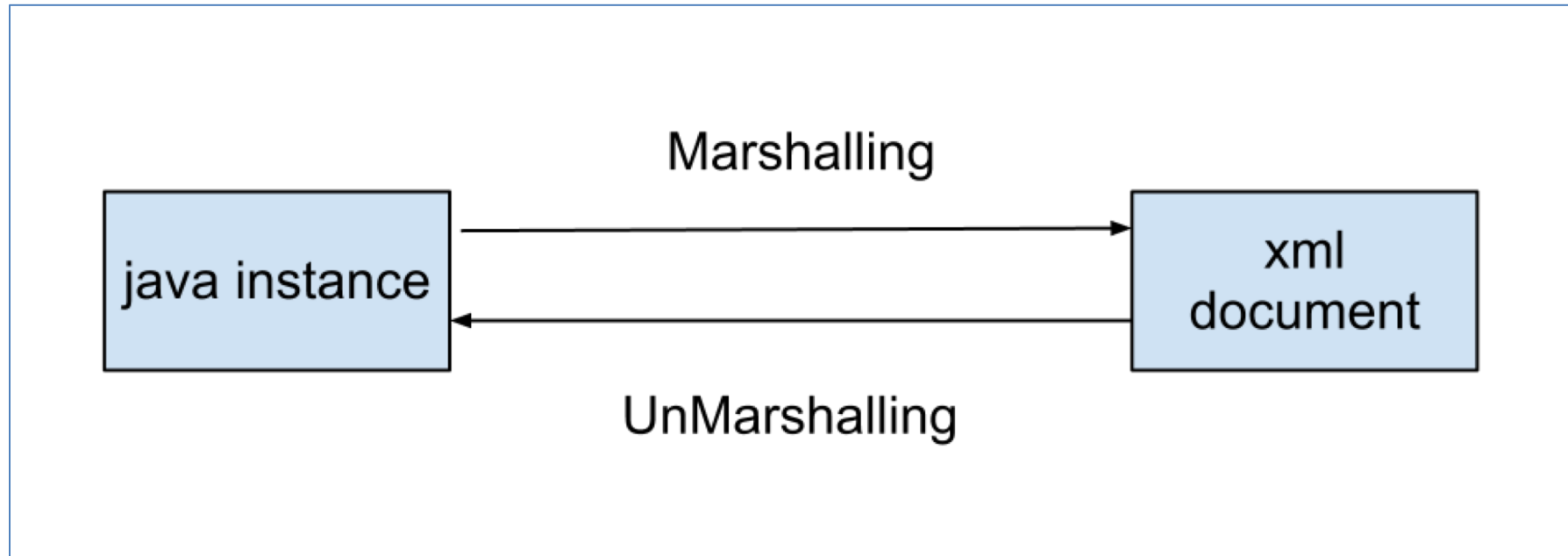


An object can be represented as a **sequence of bytes** that includes the object's **data** as well as information about the object's **type** and the **types of data stored in the object**.

# Serialization process involves **Marshalling** and **unMarshalling**

---

**Marshalling** is the process of converting the objects & data into a stream.



**UnMarshalling** is the reverse process of converting the stream back to their original objects & data.

# Questions

---

1. What does Persistence mean?
2. We use HTML for carrying data (T/F)?
3. We use XML for displaying data (T/F)?
4. XML is a replacement for HTML (T/F)?
5. When power is lost the data in a program is lost.  
What one word describes this?
6. Writing Java objects to a file is called?
7. Converting objects to a data stream is called?
8. Reading a file of data into Java objects is called?

**Any  
Questions?**

