

10.E. Almacenamiento de objetos en ficheros. Serialización.

1. Almacenamiento de objetos en ficheros. Serialización.

1.1. Serialización: utilidad.

La serialización en Java se desarrolló para utilizarse con RMI. RMI necesitaba un modo de convertir los parámetros necesarios a enviar a un objeto en una máquina remota, y también para devolver valores desde ella, en forma de flujos de bytes. Para datos primitivos es fácil, pero para objetos más complejos no tanto, y ese mecanismo es precisamente lo que proporciona la serialización.

El método `writeObject` se utiliza para guardar un objeto a través de un flujo de salida. El objeto pasado a `writeObject` debe implementar la interfaz `Serializable`.

```
FileOutputStream fisal = new FileOutputStream("cadenas.out");
```

```
ObjectOutputStream oos = new ObjectOutputStream(fisal);
```

```
Oos.writeObject();
```

La serialización de objetos se emplea también en la arquitectura de componentes software `JavaBean`. Las clases `bean` se cargan en herramientas de construcción de software visual, como `NetBeans`. Con la paleta de diseño se puede personalizar el `bean` asignando fuentes, tamaños, texto y otras propiedades.

Una vez que se ha personalizado el `bean`, para guardarlo, se emplea la serialización: se almacena el objeto con el valor de sus campos en un fichero con extensión `.ser`, que suele emplazarse dentro de un fichero `.jar`.