

FLUJOS DE SALIDA



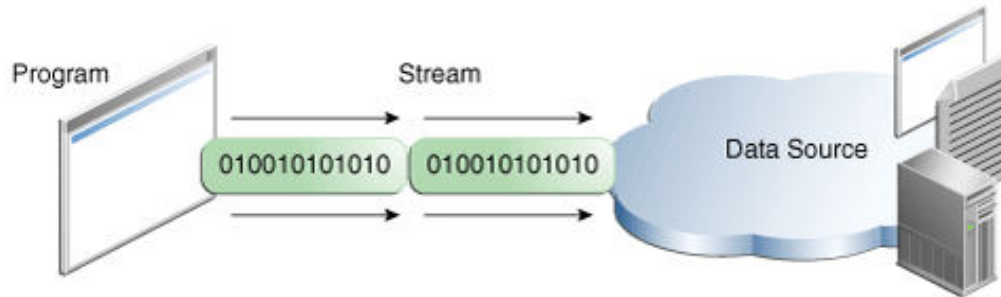
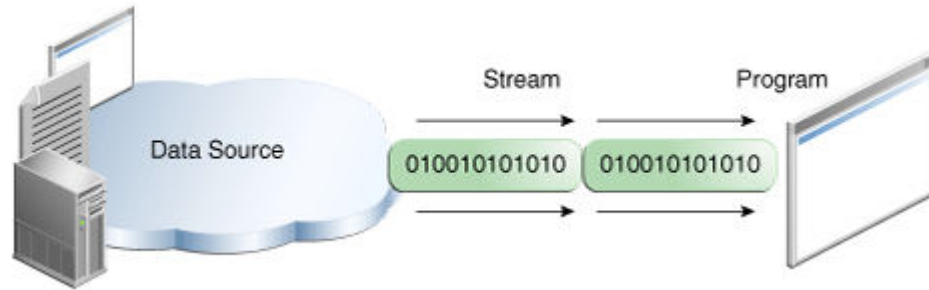
FLUJO

- ▶ Canal de comunicación de las operaciones de entrada/salida
- ▶ Literalmente, es un flujo (producción/consumo de información).

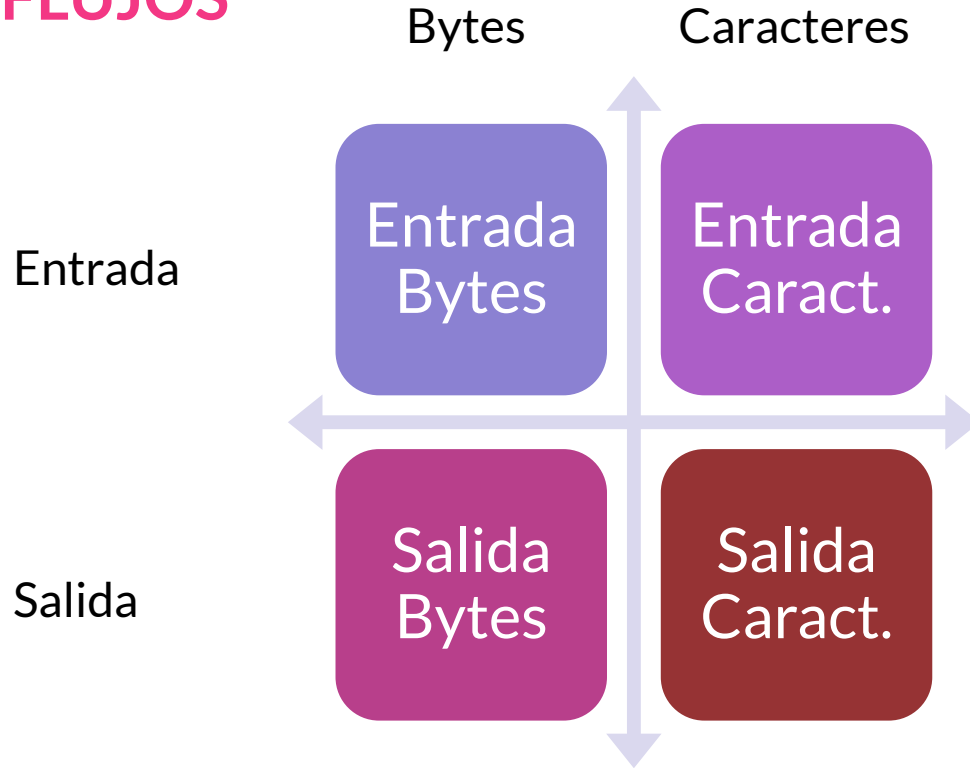
Nos da independencia:

- ▶ Entrada desde el teclado
- ▶ Salida hacia el monitor
- ▶ Lectura de un fichero
- ▶ Envío de datos por red
- ▶ ...

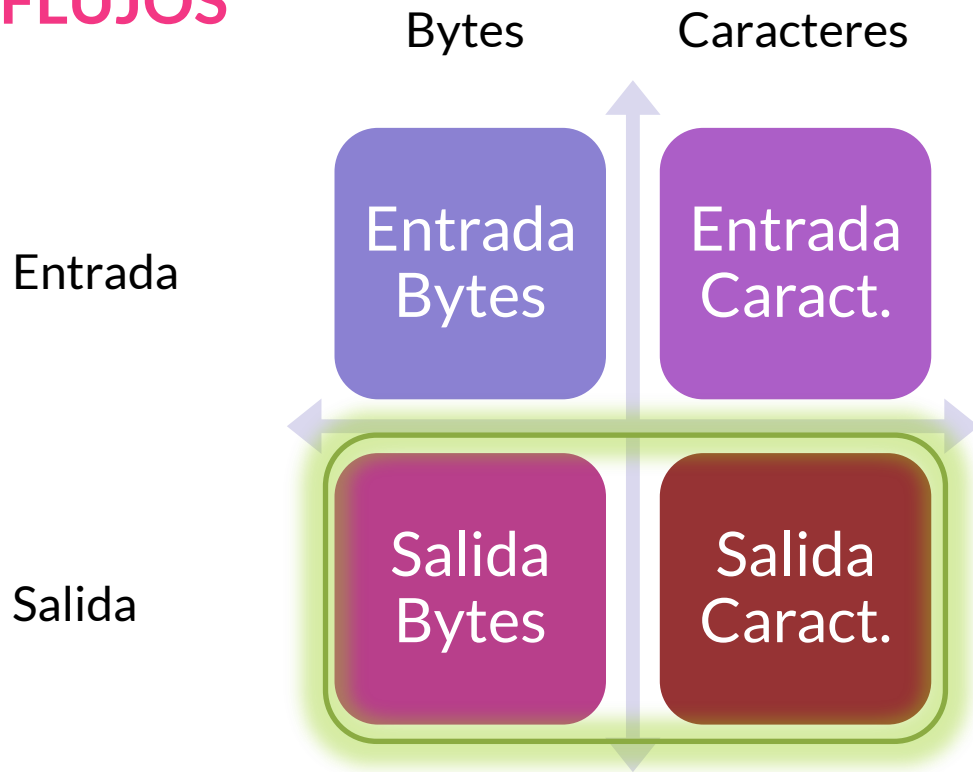
FLUJO



FLUJOS



FLUJOS



FLUJOS DE SALIDA

- ▶ Patrón básico de uso de flujos de salida:

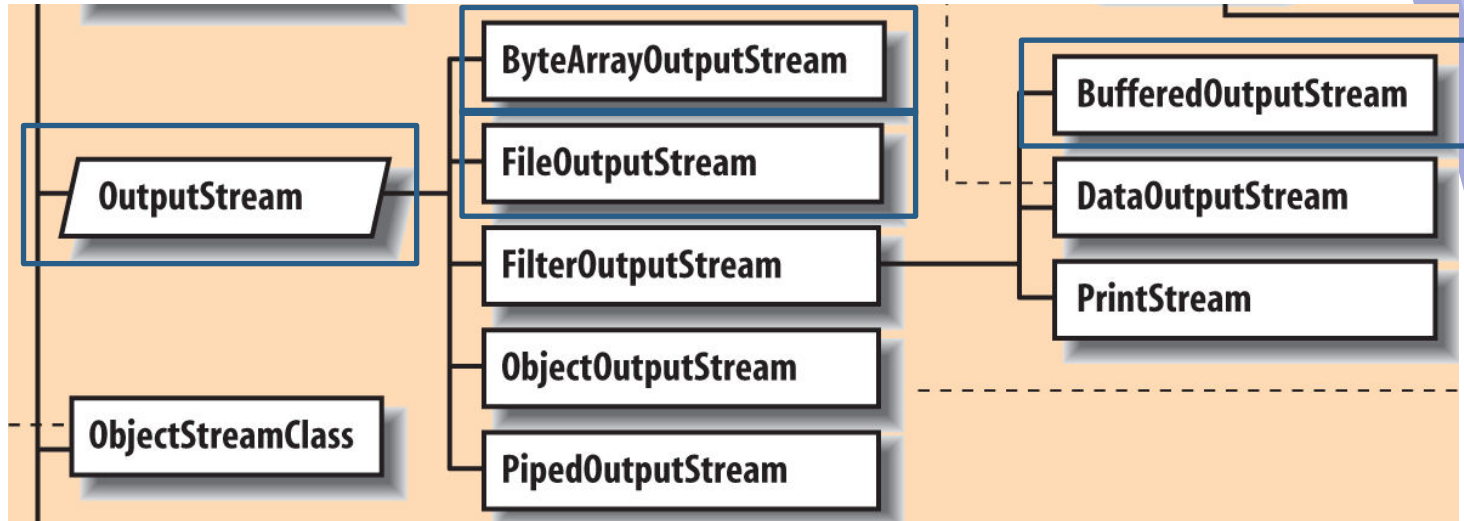
Abrir el flujo

Mientras hay datos que escribir

Escribir datos en el flujo

Cerrar el flujo

FLUJOS DE SALIDA DE BYTES

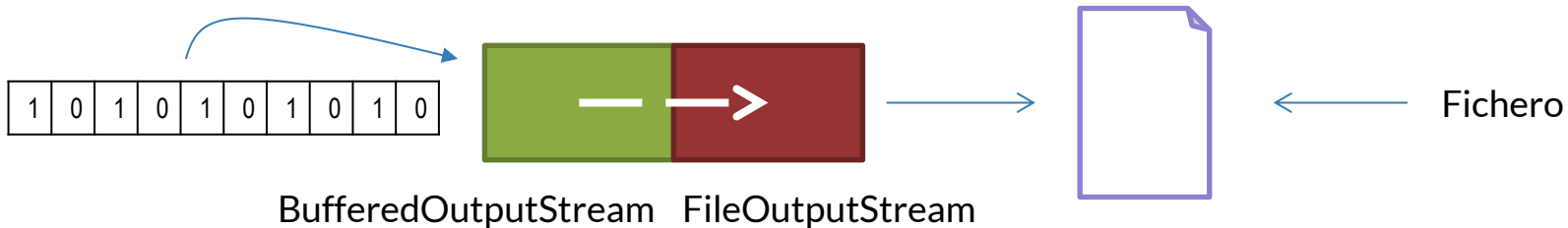


FLUJOS DE SALIDA DE BYTES

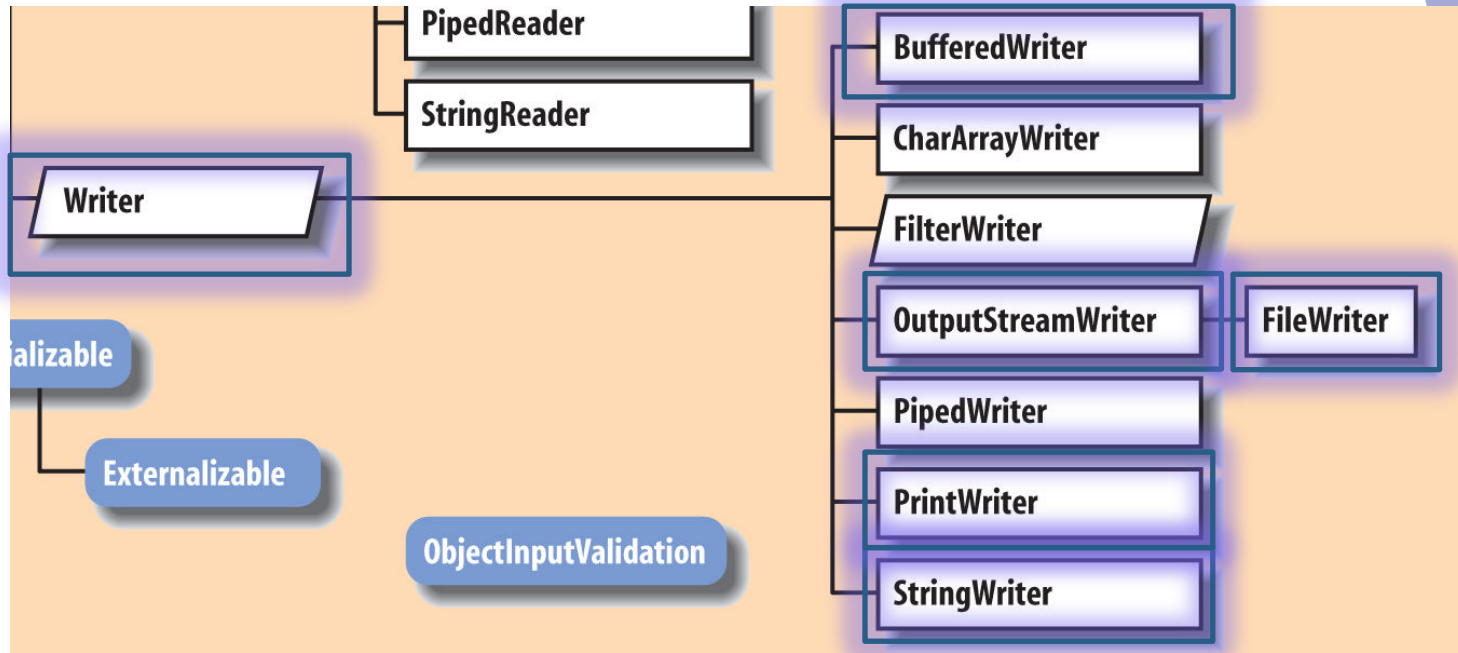
- ▶ ***OutputStream***: clase abstracta, padre de la mayoría de los flujos de bytes.
- ▶ ***FileOutputStream***: flujo que permite escribir en un fichero, byte a byte.
- ▶ ***BufferedOutputStream***: flujo que permite escribir grupos (buffers) de bytes.
- ▶ ***ByteArrayOutputStream***: flujo que permite escribir *en memoria*, obteniendo lo escrito en un array de bytes.

FLUJOS HACIA OTROS FLUJOS

- ▶ Solo *FileOutputStream* tiene un constructor que acepta una ruta (entre otras opciones).
- ▶ El resto reciben en sus constructores *OutputStream*. ¿Por qué?
- ▶ Podemos construir flujos que escriben en flujos (encadenados).



FLUJOS DE SALIDA DE CARACTERES



FLUJOS DE SALIDA DE CARACTERES

- ▶ **Writer**: clase abstracta, padre de la mayoría de los flujos de caracteres.
- ▶ **FileWriter**: flujo que permite escribir en un fichero, caracter a caracter.
- ▶ **BufferedWriter**: flujo que permite escribir líneas de texto.
- ▶ **StringWriter**: flujo que permite escribir *en memoria*, obteniendo lo escrito en un String
- ▶ **OutputStreamWriter**: flujo que permite transformar un OutputStream en un Writer.
- ▶ **PrintWriter**: flujo que permite escribir tipos básicos Java.