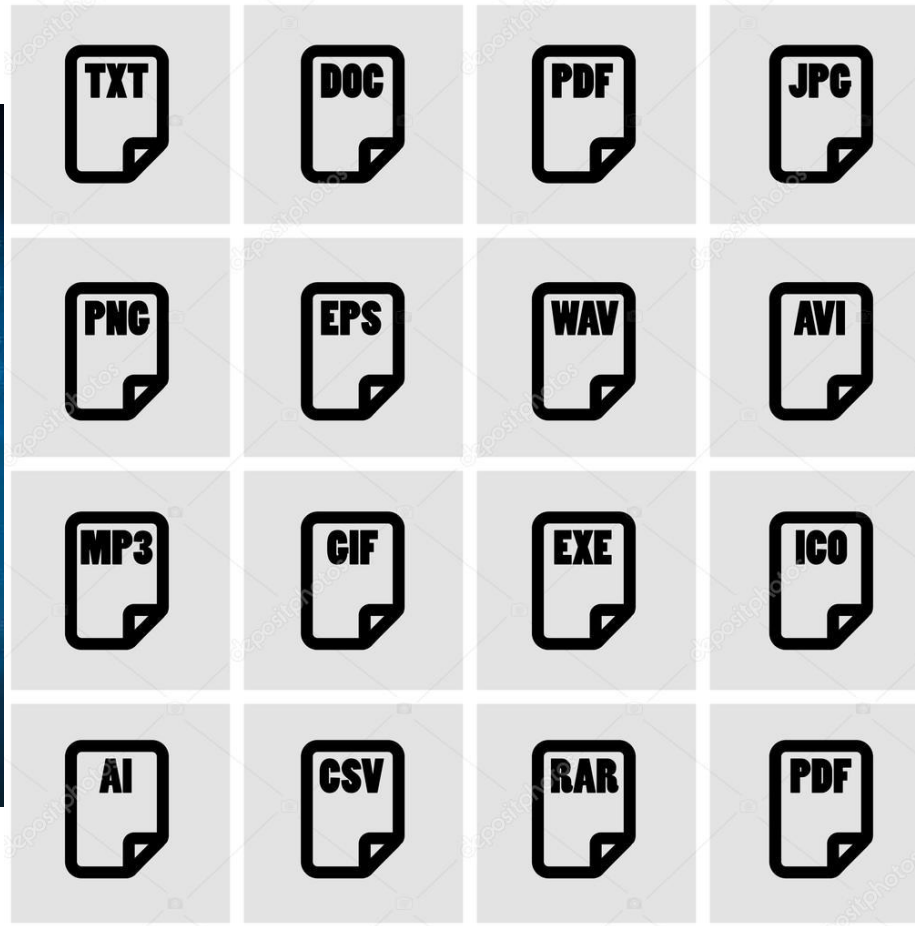


# Trabajar con Archivos - Namespace I-O



# Trabajar con Archivos - Namespace I-O



# System.IO - Clases

## BinaryReader

- Lee datos primitivos de una secuencia binaria

## BinaryWriter

- Escribe datos primitivos en formato binario

## BufferedStream

- Es un almacenamiento temporal para una secuencia de bytes

## Directory

- Ayuda a manipular una estructura de directorio

## DirectoryInfo

- Se utiliza para realizar operaciones en directorios

## DriveInfo

- Proporciona información de las unidades de disco

## File

- Ayuda en la manipulación de archivos

## FileInfo

- Se utiliza para realizar operaciones en archivos

# System.IO - Clases

## FileStream

- Se usa para leer y escribir en cualquier ubicación de un archivo

## MemoryStream

- Se utiliza para acceso aleatorio a datos almacenados en la memoria

## Path

- Realiza operaciones en la información de la ruta de un archivo

## StreamReader

- Se utiliza para leer caracteres de una secuencia de bytes

## StreamWriter

- Se utiliza para escribir caracteres en una secuencia

## StringReader

- Se utiliza para leer desde un buffer de cadena

## StringWriter

- Se utiliza para escribir en un buffer de cadena

# FileStream

## Estructura Básica de FileStream

```
FileStream <nombre_objeto> = new FileStream( <nombre_archivo>, <Enumerador FileMode>,  
    <Enumerador FileAccess>, <Enumerador FileShare>);
```

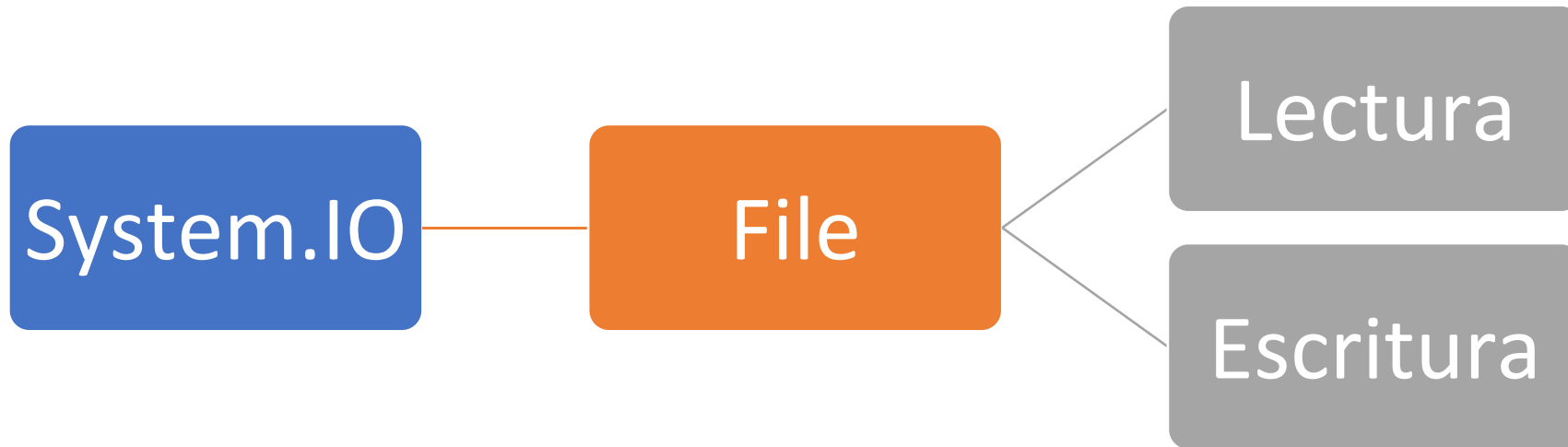
## Ejemplo de FileStream

```
FileStream F = new FileStream("ejemplo.txt", FileMode.Open, FileAccess.Read, FileShare.Read);
```

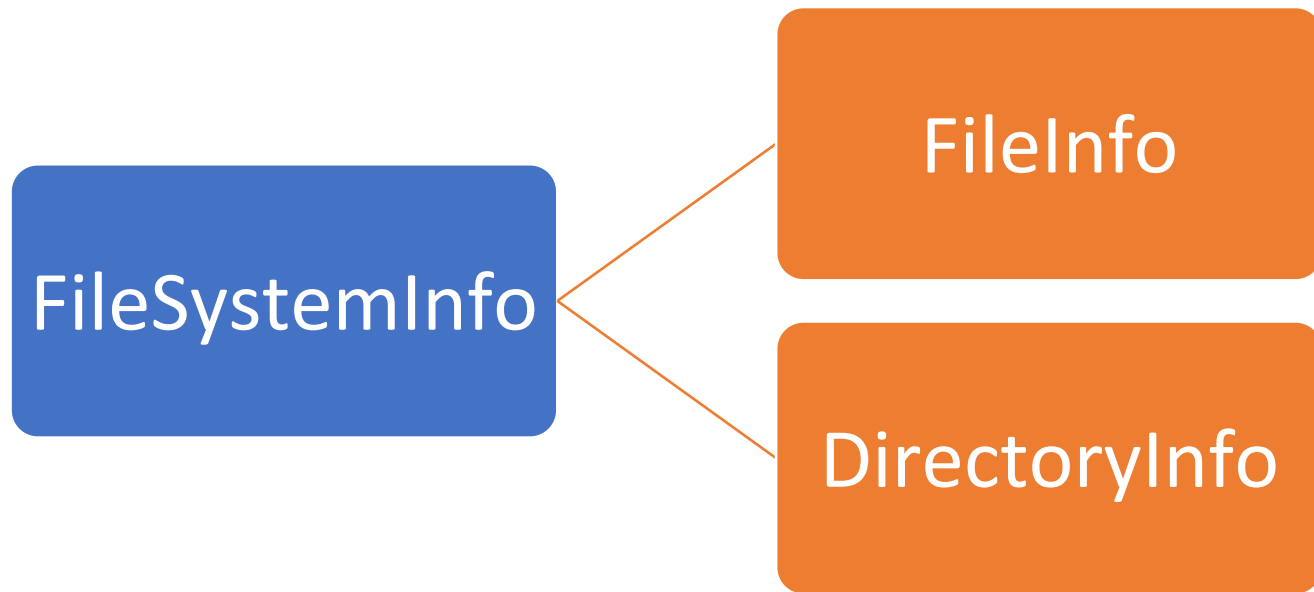
# FileStream - Parámetros



# Operaciones Básicas con Archivos y Directorios



# *Sistema de Ficheros de Windows*





# DirectoryInfo - Propiedades

## Attributes

- Obtiene los atributos del archivo o directorio actual

## CreationTime

- Obtiene la hora de creación del archivo o directorio actual

## Exists

- Obtiene un valor booleano que indica si el directorio existe

## Extension

- Obtiene la cadena que representa la extensión del archivo

## FullName

- Obtiene la ruta completa del directorio o archivo

## LastAccessTime

- Obtiene la hora a la que se accedió por última vez al archivo o directorio actual

## Name

- Obtiene el nombre de esta instancia de DirectoryInfo

# DirectoryInfo - Métodos

1

- **public void Create ()**
- Crea un directorio

2

- **public Directory CreateSubdirectory (string path)**
- Crea un subdirectorio o subdirectorios en la ruta especificada. La ruta especificada puede ser relativa a esta instancia de la clase DirectoryInfo

3

- **public override void Delete ()**
- Elimina este DirectoryInfo si está vacío

4

- **public DirectoryInfo [] GetDirectories ()**
- Devuelve los subdirectorios del directorio actual

5

- **public FileInfo [] GetFiles ()**
- Devuelve una lista de archivos del directorio actual

# FileInfo - Propiedades

Attributes	<ul style="list-style-type: none"><li>• Obtiene los atributos del archivo actual</li></ul>
CreationTime	<ul style="list-style-type: none"><li>• Obtiene la hora de creación del archivo actual</li></ul>
Directory	<ul style="list-style-type: none"><li>• Obtiene una instancia del directorio al que pertenece el archivo</li></ul>
Exists	<ul style="list-style-type: none"><li>• Obtiene un valor booleano que indica si el archivo existe</li></ul>
Extension	<ul style="list-style-type: none"><li>• Obtiene la cadena que representa la extensión del archivo</li></ul>
FullName	<ul style="list-style-type: none"><li>• Obtiene la ruta completa del archivo</li></ul>
LastAccessTime	<ul style="list-style-type: none"><li>• Obtiene la hora a la que se accedió por última vez al archivo actual</li></ul>
LastWriteTime	<ul style="list-style-type: none"><li>• Obtiene la hora de la última actividad de escritura del archivo</li></ul>
Length	<ul style="list-style-type: none"><li>• Obtiene el tamaño, en bytes, del archivo actual</li></ul>
Name	<ul style="list-style-type: none"><li>• Obtiene el nombre del archivo</li></ul>

# FileInfo - Métodos

1

- **public StreamWriter AppendText ()**
- Crea un StreamWriter que agrega texto al archivo representado por esta instancia de FileInfo

2

- **public FileStream Create ()**
- Crea un archivo

3

- **public override void Delete ()**
- Elimina un archivo de forma permanente

4

- **public void MoveTo (string destFileName)**
- Mueve un archivo especificado a una nueva ubicación, brindando la opción de especificar un nuevo nombre de archivo

5

- **public FileStream Open (FileMode mode)**
- Abre un archivo en el modo especificado

6

- **public FileStream Open (FileMode mode, FileAccess access)**
- Abre un archivo en el modo especificado con lectura, escritura o acceso de lectura / escritura

7

- **public FileStream Open (FileMode mode, FileAccess access, FileShare share)**
- Abre un archivo en el modo especificado con lectura, escritura o acceso de lectura / escritura y la opción de compartir especificada

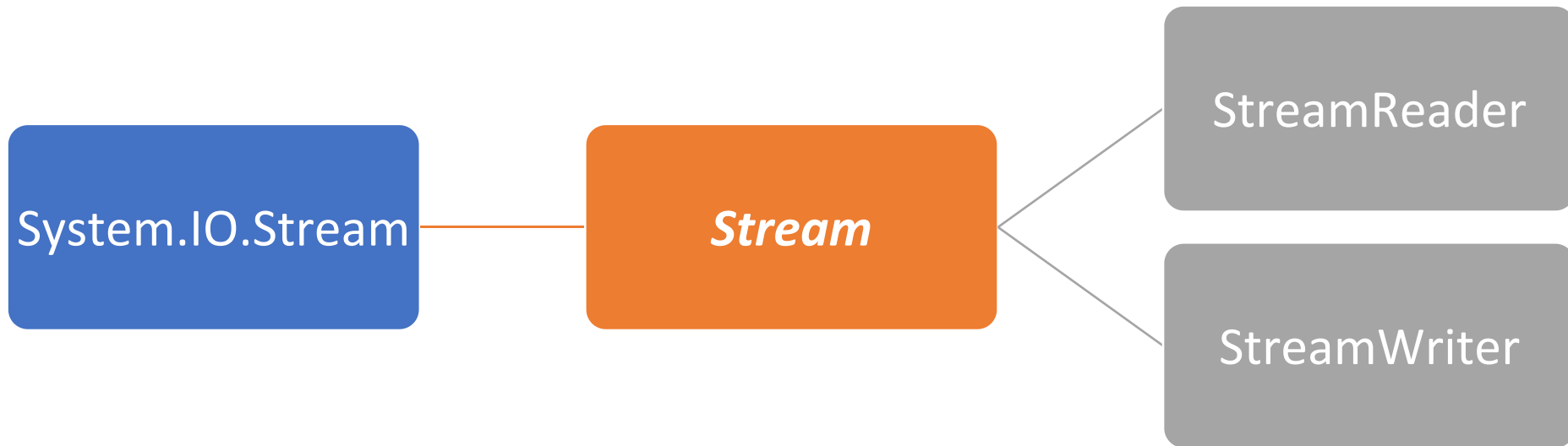
8

- **public FileStream OpenRead ()**
- Crea un FileStream de solo lectura

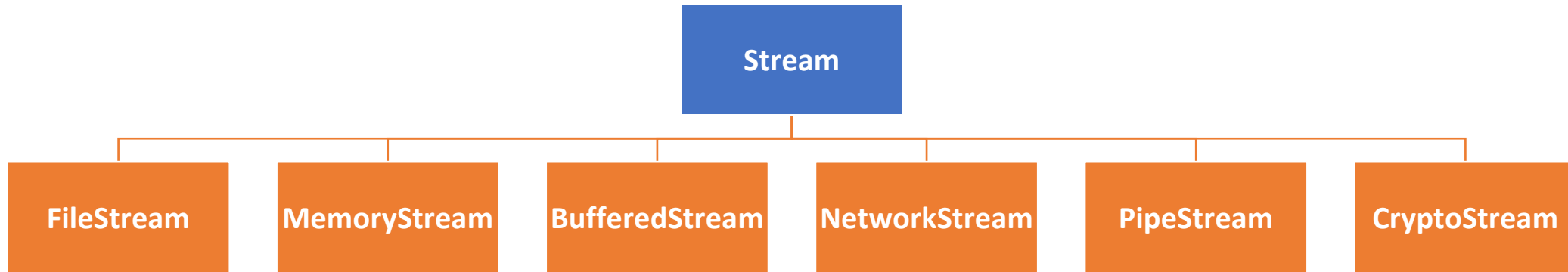
9

- **public FileStream OpenWrite ()**
- Crea un FileStream de solo escritura

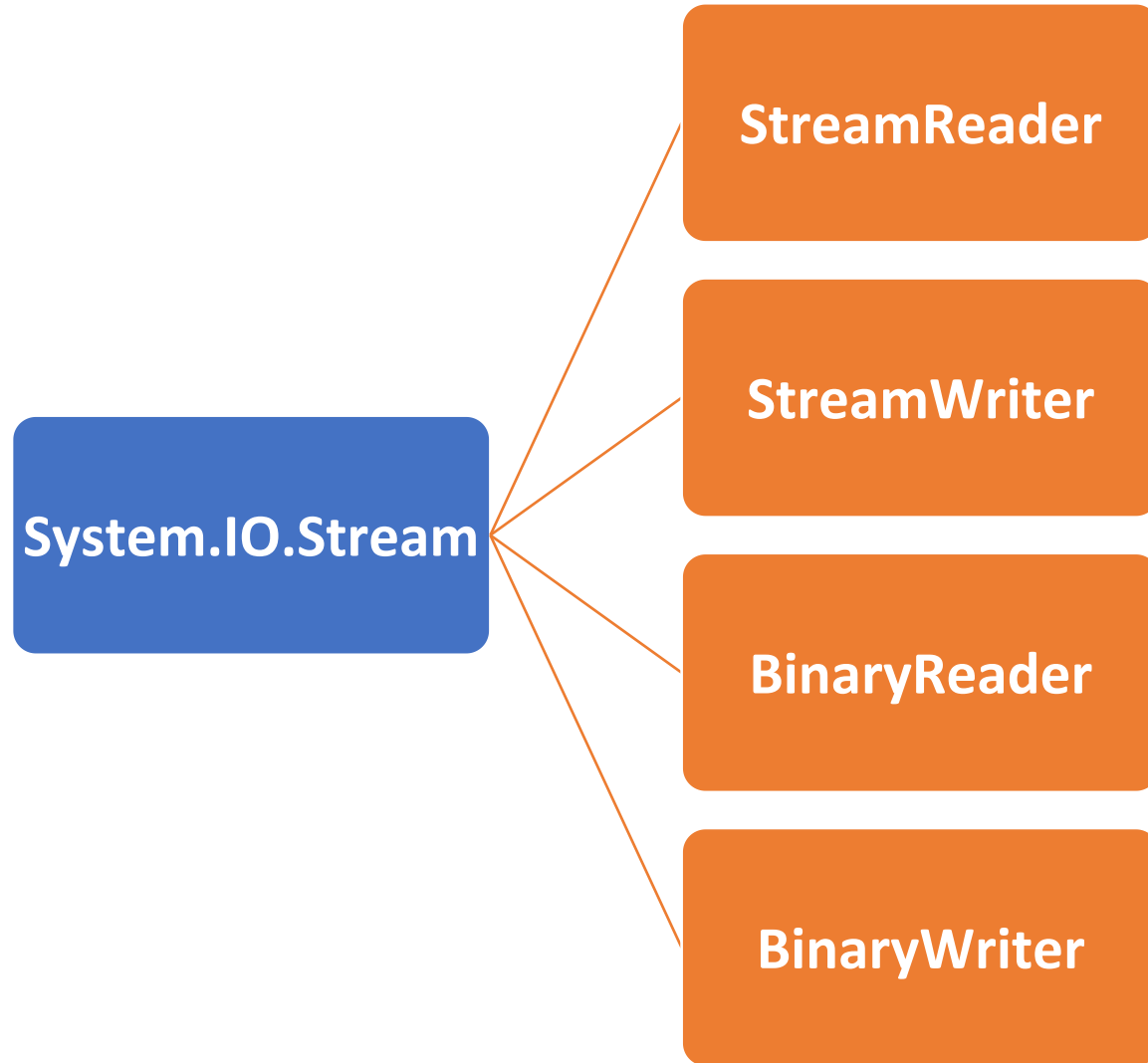
# La Clase Stream



# La Clase Stream



# Readers y Writers



# Readers y Writers

Cadena Lectura/Escritura

Lectura/Escritura de bytes

StreamReader  
StreamWriter

FileStream



Archivo Físico  
"prueba.txt"



# StreamReader - Métodos

1

- **public override void Close()**
- Cierra el objeto StreamReader y la secuencia subyacente, y libera todos los recursos del sistema asociados con las operaciones de lectura.

2

- **public override int Peek()**
- Devuelve el siguiente carácter disponible pero no lo consume

3

- **public override int Read()**
- Lee el siguiente carácter de la secuencia de entrada y avanza la posición del carácter en uno

# StreamWriter – Métodos

1

- **public override void Close()**
- Cierra el objeto StreamWriter actual y la secuencia subyacente

2

- **public override void Flush()**
- Borra todos los almacenamientos intermedios de la operación de escritura actual y hace que los datos almacenados en el búfer se escriban en el flujo subyacente

3

- **public virtual void Write(bool value)**
- Escribe la representación de texto de un valor booleano en la secuencia de texto o secuencia

4

- **public override void Write(char value)**
- Escribe un carácter en la transmisión

5

- **public virtual void Write(decimal value)**
- Escribe la representación de texto de un valor decimal en la secuencia de texto o secuencia

# StreamWriter – Métodos

6

- **public virtual void Write(double value)**
- Escribe la representación de texto de un valor de coma flotante de 8 bytes en la secuencia de texto o secuencia

7

- **public virtual void Write(int value)**
- Escribe la representación de texto de un entero de 4 bytes con signo en la secuencia de texto o secuencia

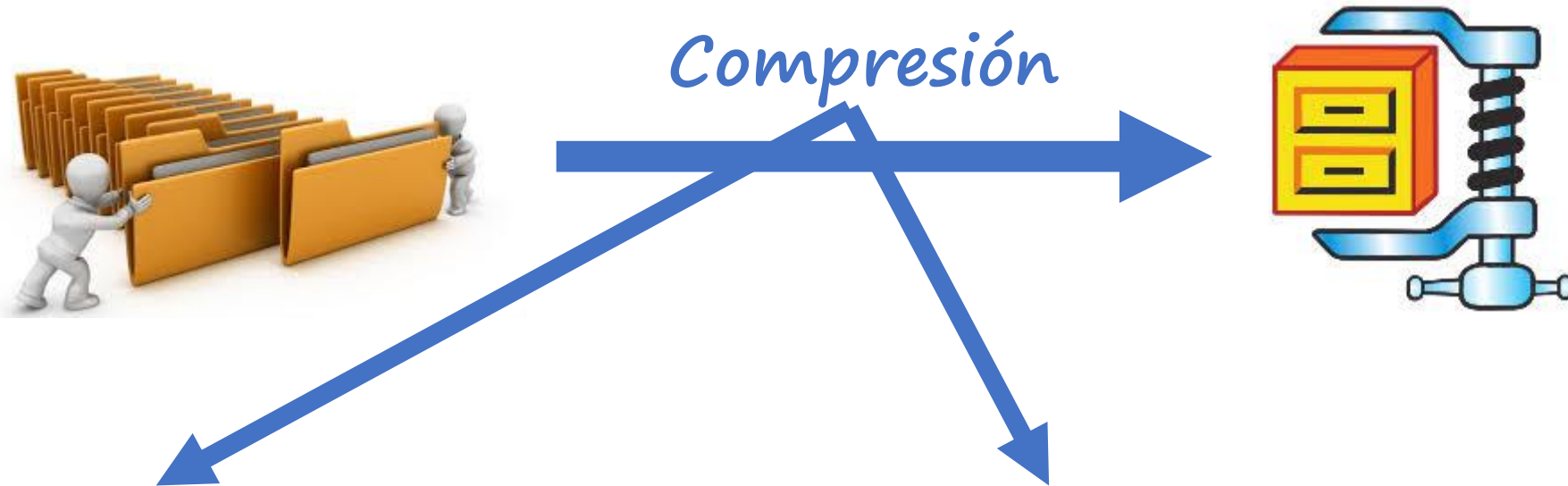
8

- **public override void Write(string value)**
- Escribe una cadena en la secuencia

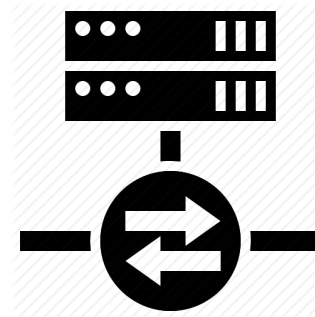
9

- **public virtual void WriteLine()**
- Escribe un terminador de línea en la secuencia de texto o secuencia

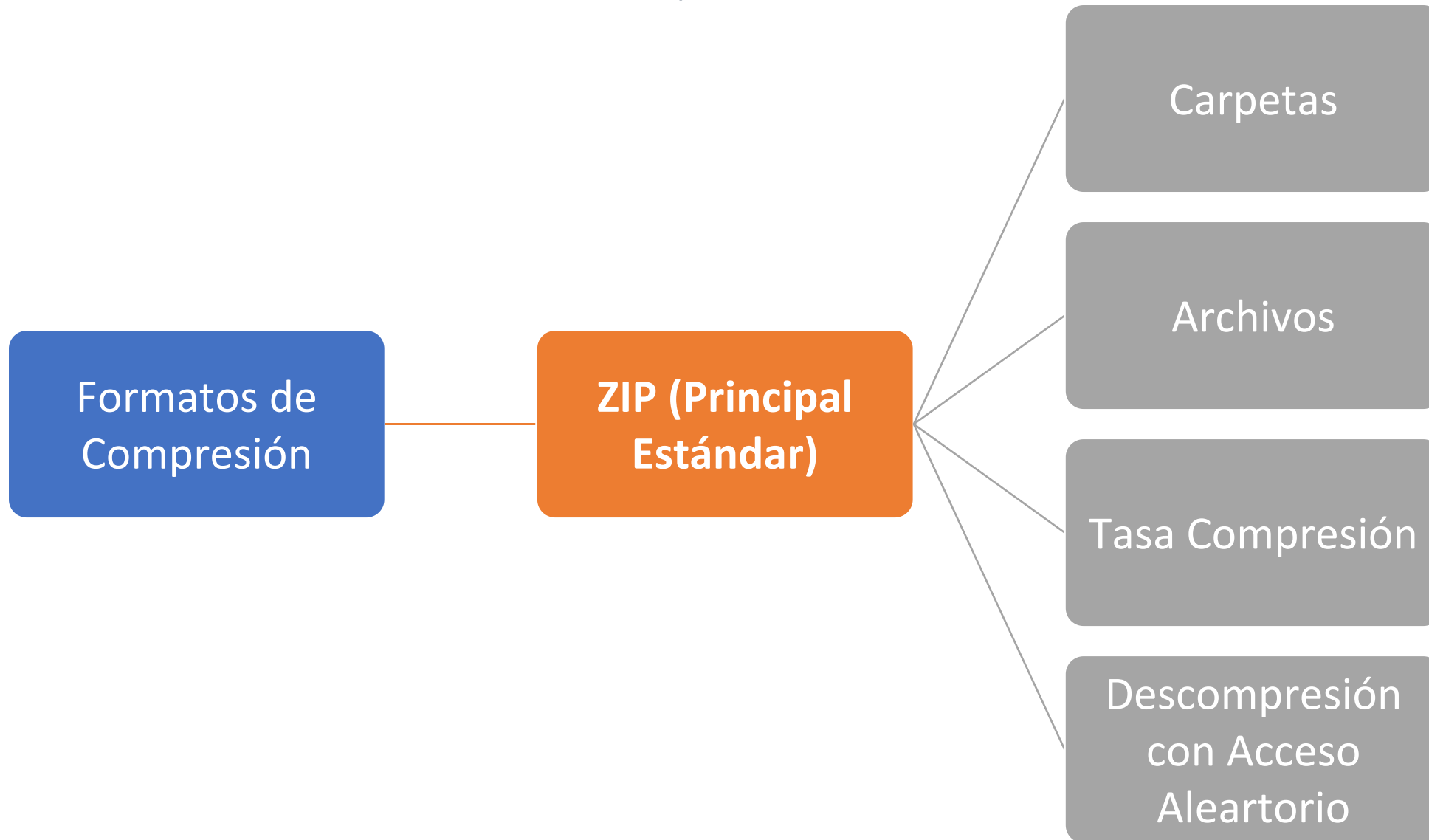
# Trabajando con Archivos ZIP



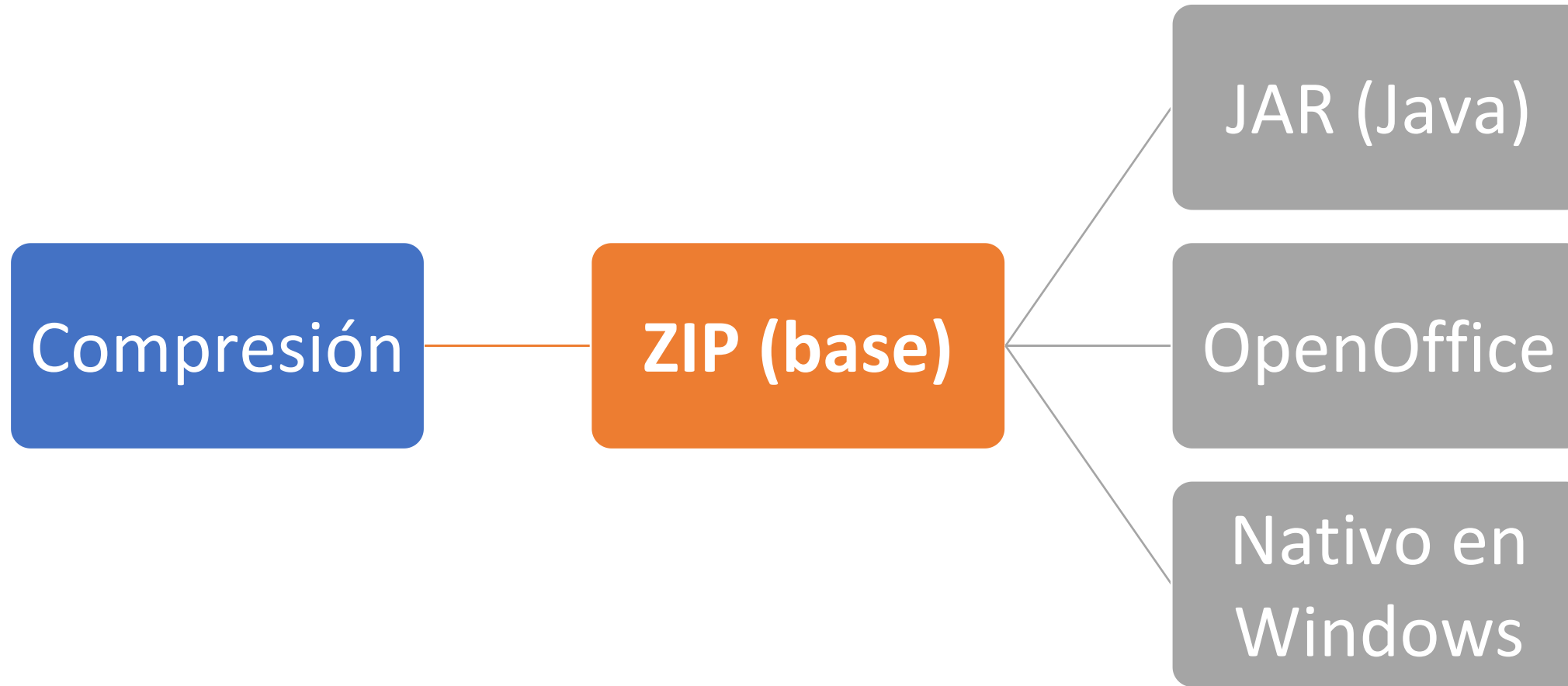
Ahorro en Almacenamiento Menor Consumo de Red



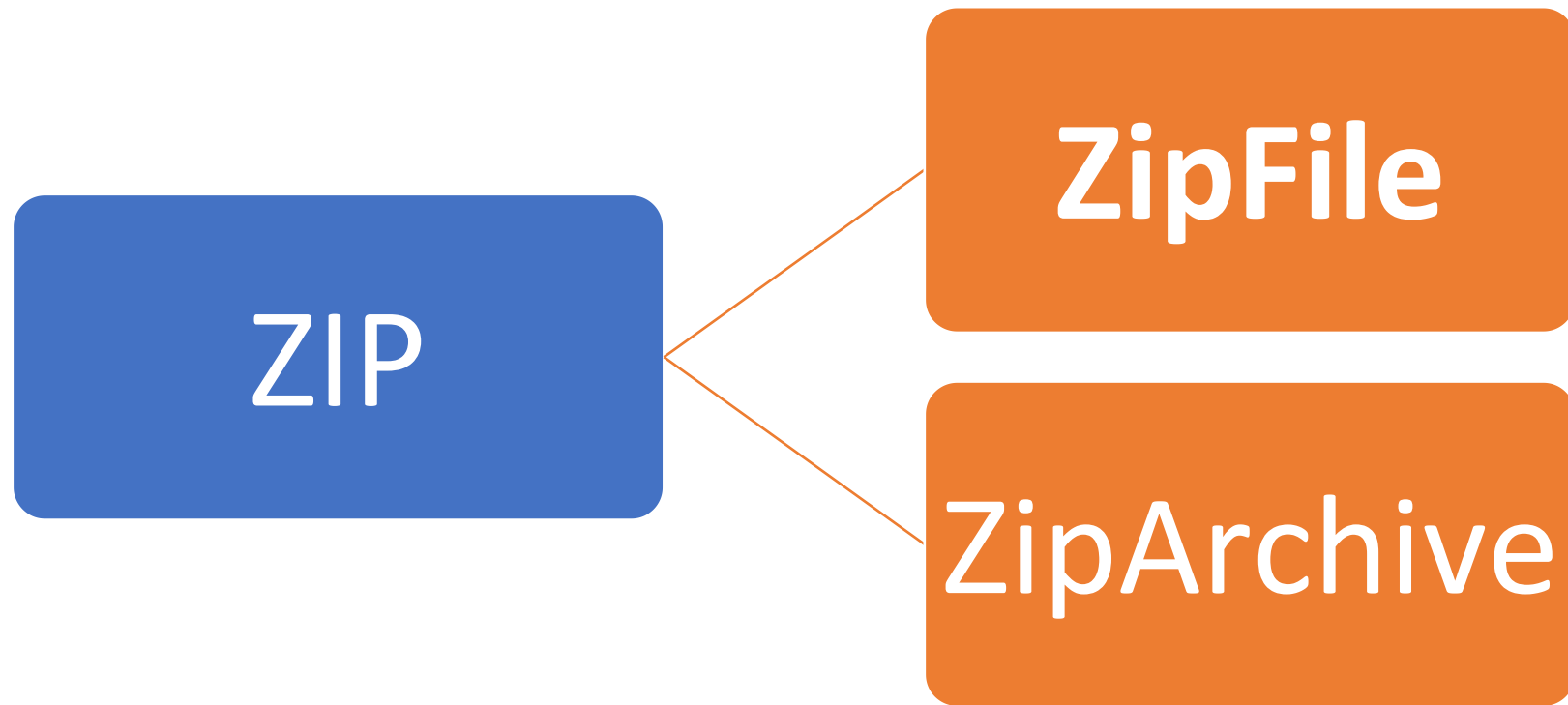
# Compresión - ZIP



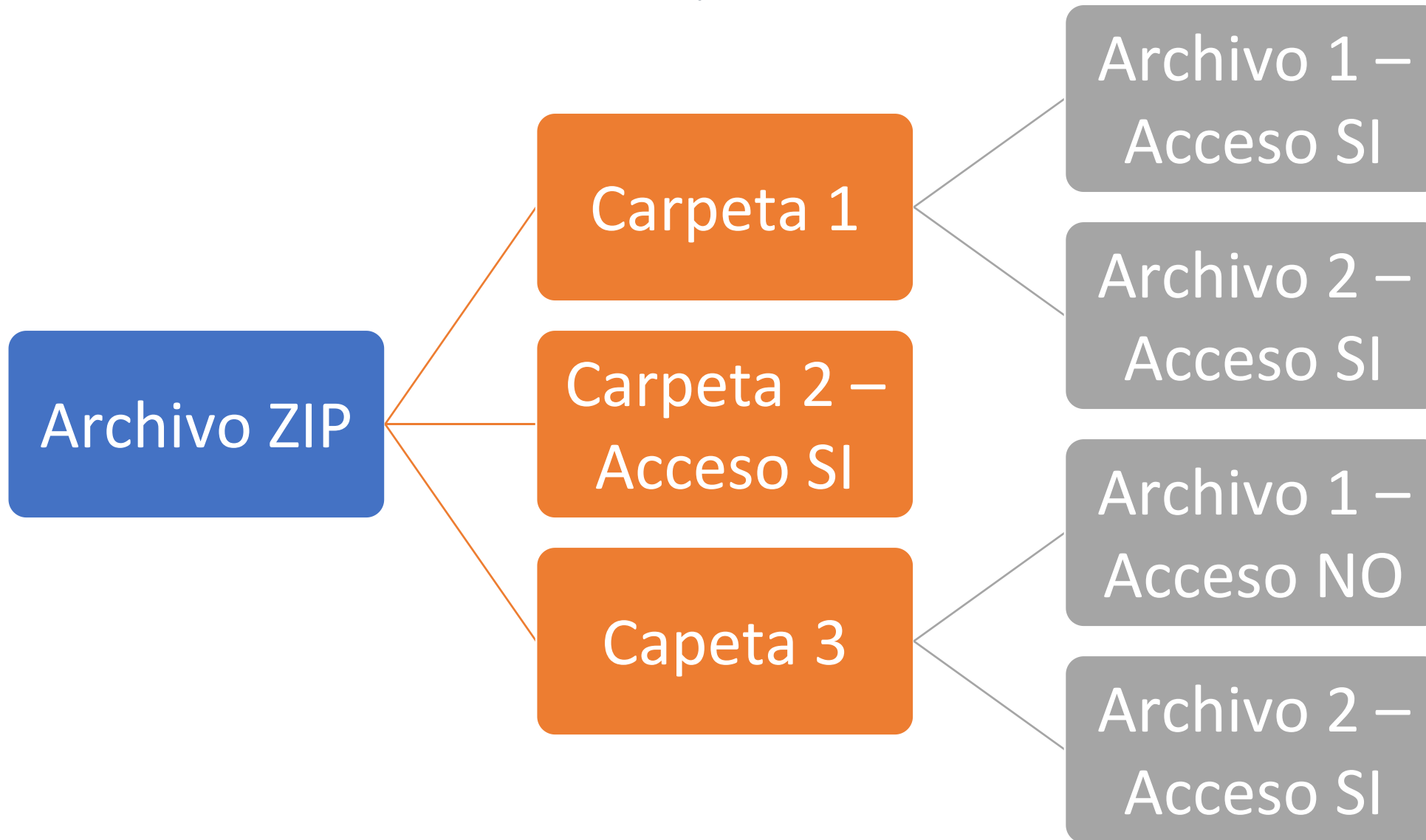
# *Características- ZIP*



# Trabajando con ZIP

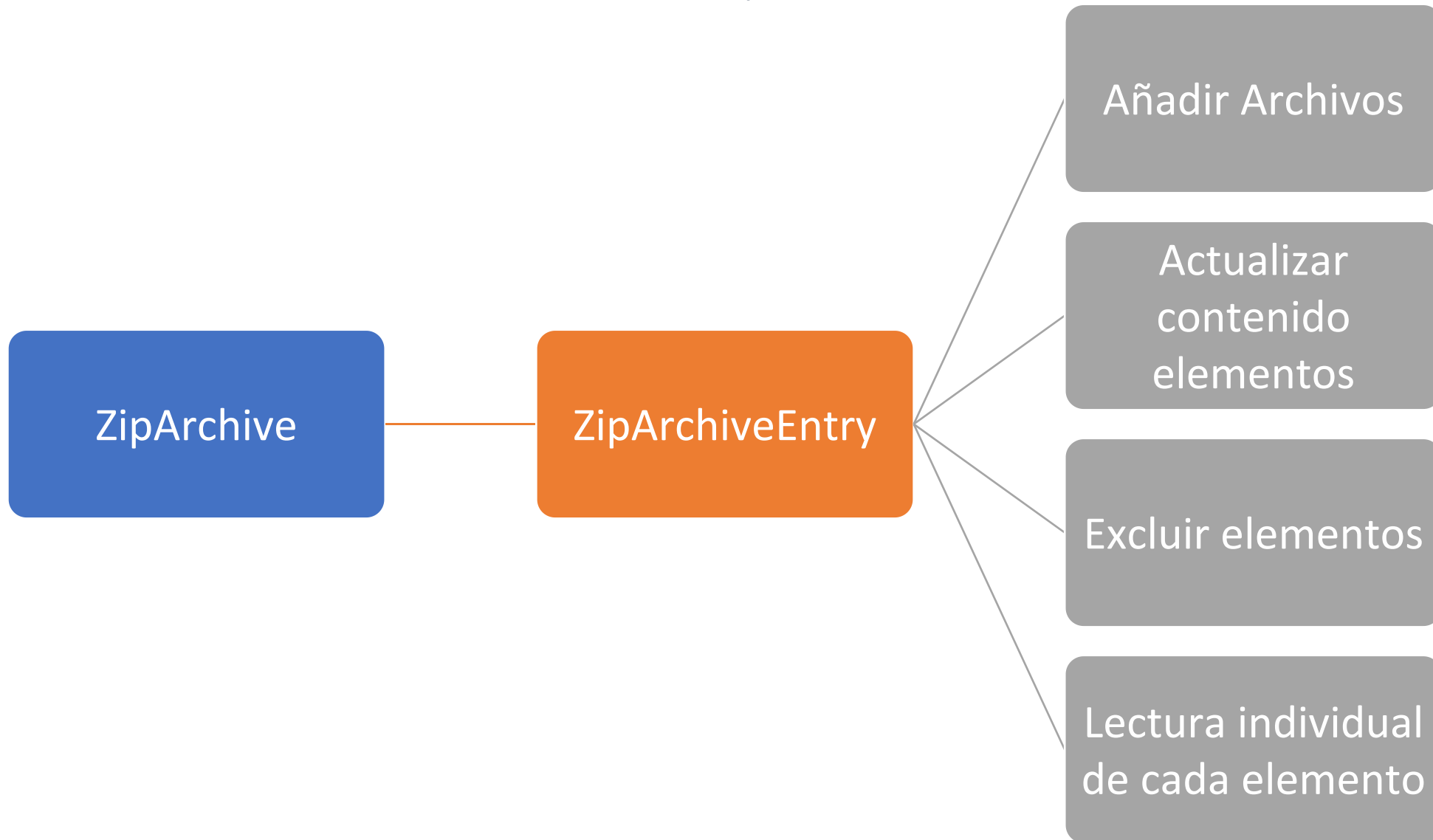


# ZipArchive





# ZipArchive



# ZipArchive

