

User - Space

SYSTEM - CALL

Kernel - Space

2.1.3 Las llamadas al sistema

Interfaz fundamental entre las aplicaciones y el kernel del sistema operativo



Definición y concepto fundamental

↔ ¿Qué son las llamadas al sistema?

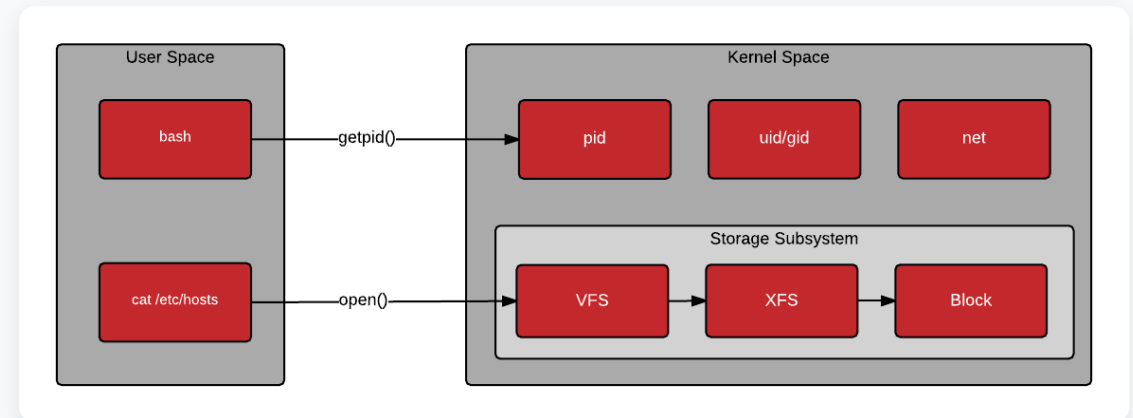
Conjunto de instrucciones que sirven como **interfaz con el sistema operativo** para solicitar servicios

↕ Mecanismo principal

- Programas en espacio de usuario solicitan servicios al kernel
- Transferencia temporal de control al sistema operativo
- Ejecución con privilegios elevados

🛡 Seguridad y estabilidad

- Los programas no acceden directamente al hardware
- Control centralizado de recursos críticos
- Devolución de resultados al programa solicitante



Función y propósito



Capa de abstracción

Entre programas de usuario y hardware subyacente



Seguridad del sistema

Controla el acceso a recursos críticos



Interfaz estandarizada

Permite funcionamiento en diferentes configuraciones de hardware



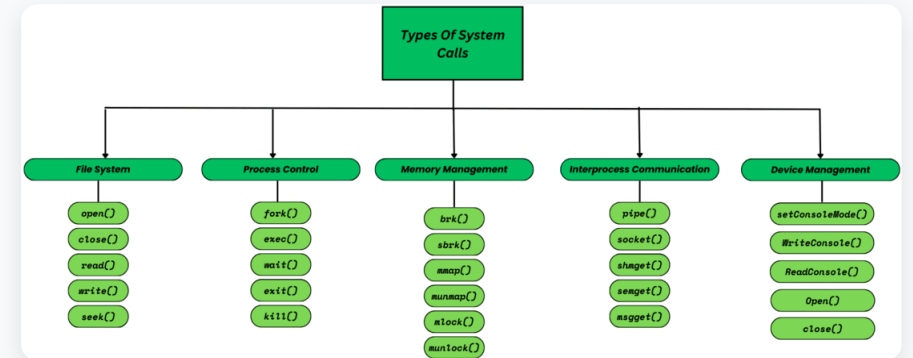
Portabilidad

Facilita la ejecución entre diferentes sistemas operativos



Control de recursos

Evita que un programa monopolice recursos del sistema



Tipos de llamadas al sistema

El sistema operativo ofrece diferentes tipos de llamadas para gestionar los recursos y servicios del sistema



Gestión de procesos

Creación, terminación y control de procesos



Gestión de memoria

Asignación, liberación y mapeo de memoria



Gestión de archivos

Operaciones de lectura, escritura y manipulación



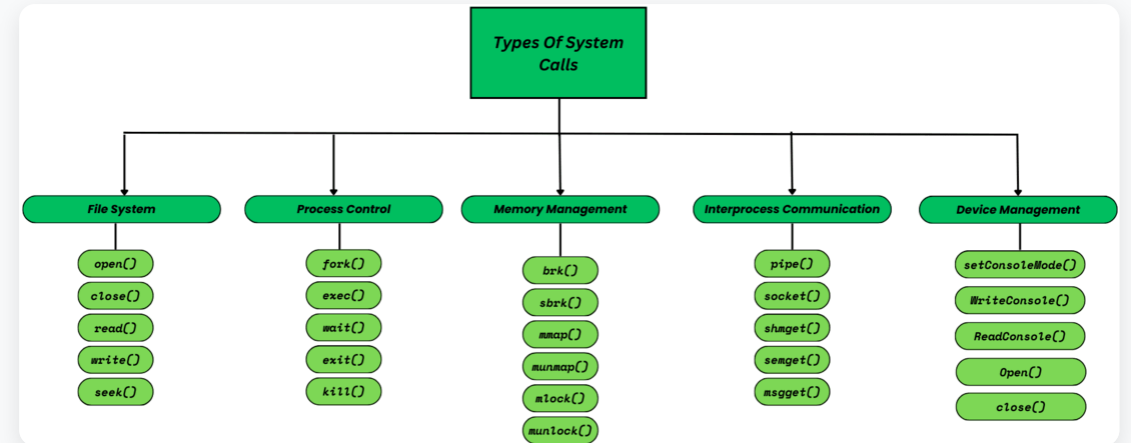
Protección y seguridad

Control de acceso y permisos de recursos



Gestión de dispositivos E/S

Control de dispositivos y manejo de interrupciones



Llamadas para gestión de procesos

⚙ Operaciones principales



Crear un proceso



Terminar un proceso



Cambiar prioridades



Esperar proceso hijo



Cargar programas



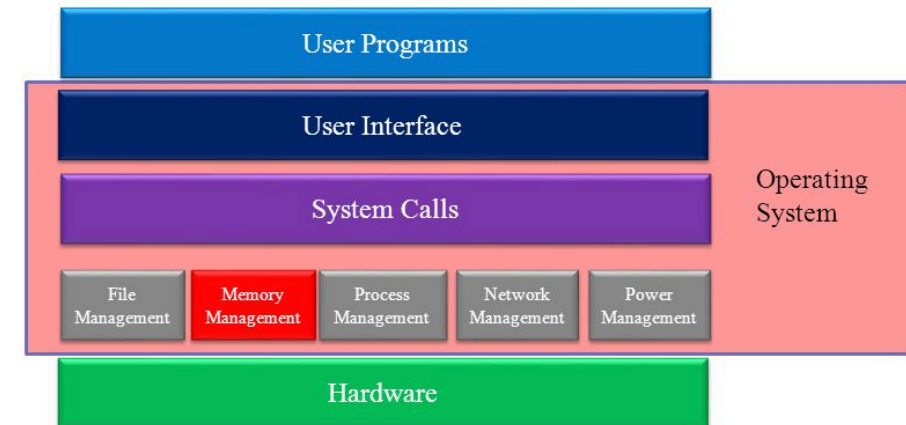
Ejecutar programas

🏠 Relación padre-hijo

! Importancia

- Un proceso es creado por otro mediante llamadas al sistema
- Se establece una **relación jerárquica**
- Fundamental para la **multitarea**
- Permite ejecución **simultánea** de aplicaciones

Memory Management in OS



Llamadas para gestión de memoria

🔧 Operaciones principales



Solicitar memoria
(malloc, calloc)



Liberar memoria (free)



Mapear archivos en
memoria

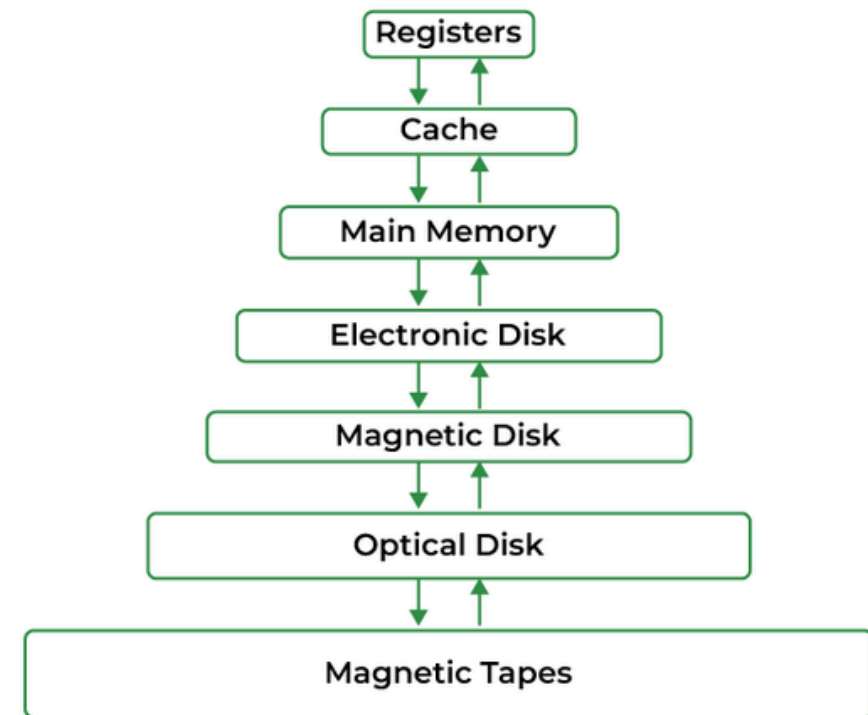


Configurar memoria
virtual

📌 Importancia

! Rendimiento del sistema

- Gestión **eficiente** de recursos
- Crucial en entornos **multiproceso**
- Evita **fugas de memoria**
- Optimiza el uso de **recursos limitados**



Llamadas para gestión de archivos

📁 Operaciones principales



Crear, eliminar y renombrar



Abrir y cerrar archivos



Leer y escribir datos



Obtener información



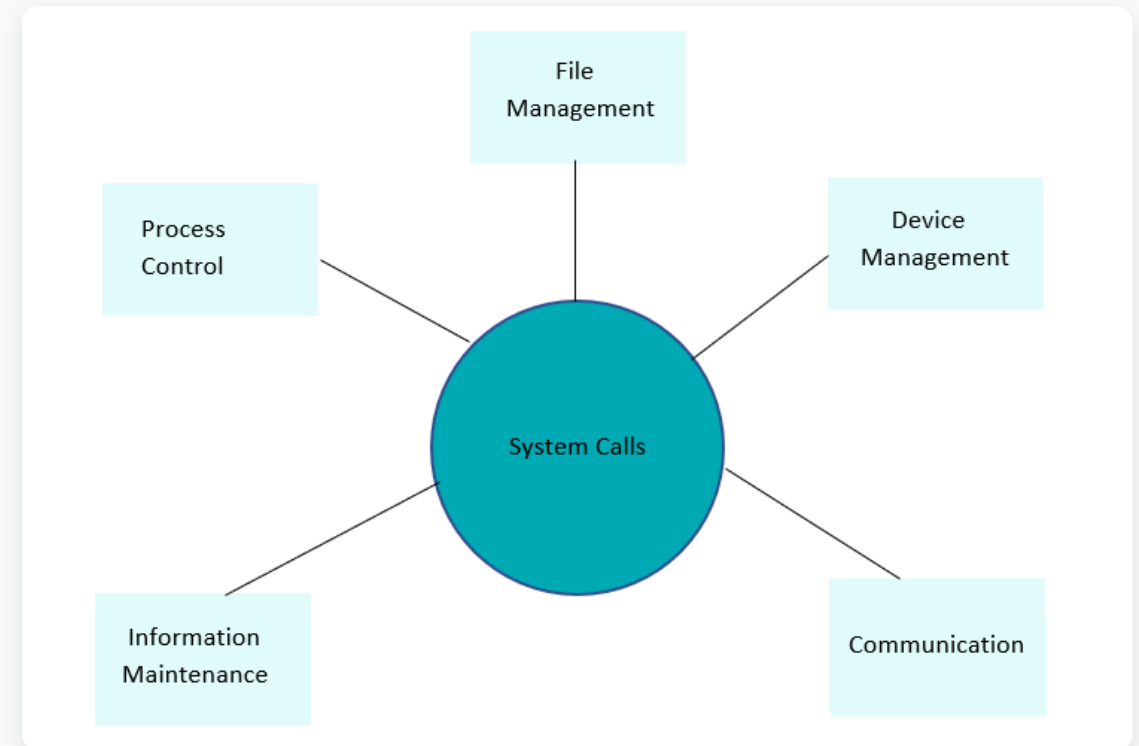
Cambiar permisos

🔍 Abstracción uniforme



Ventajas

- Acceso transparente a **diferentes sistemas** de archivos
- Independencia del **hardware** subyacente
- Interfaz **estandarizada** para aplicaciones
- Facilita la **portabilidad** entre sistemas



Llamadas para protección y seguridad

🛡️ Funciones de seguridad

👤 Gestión de propietarios

🔒 Permisos de acceso

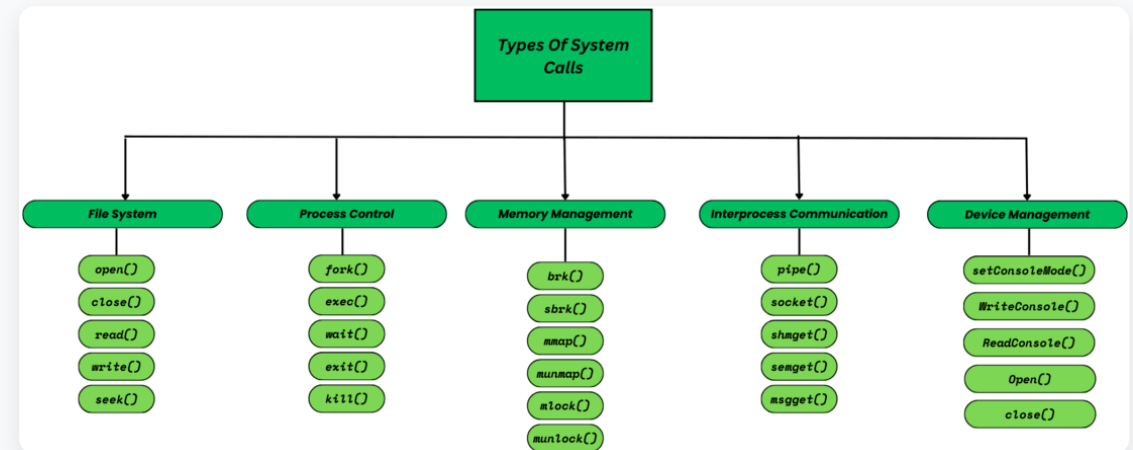
✅ Autenticación de usuarios

🛡️ Control de acceso

✅ Importancia del sistema

! Protección integral

- Seguridad para **usuarios**
- Protección de **software**
- Resguardo de **información**
- Control de **recursos críticos**



Llamadas para gestión de dispositivos de E/S

📁 Operaciones principales

⚙️ Configurar dispositivos

➡️ Enviar comandos

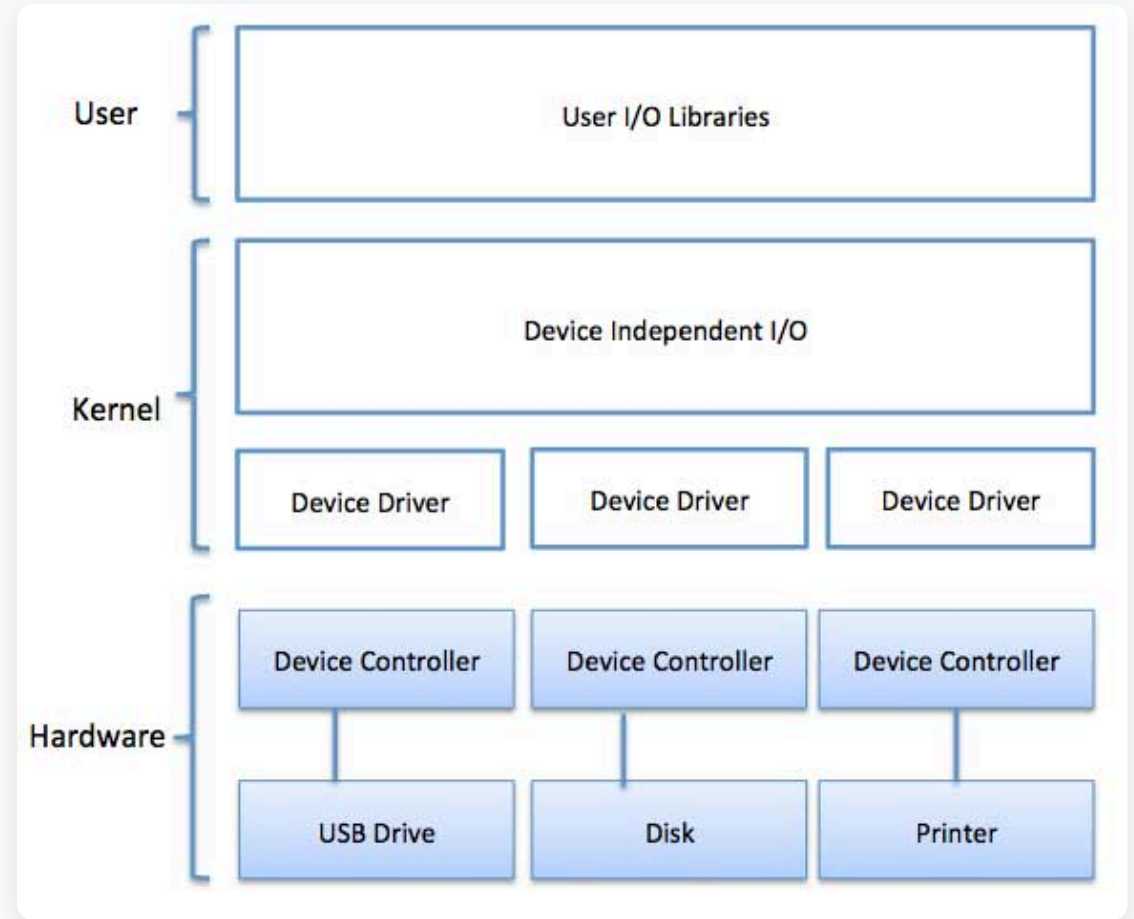
↕️ Leer y escribir datos

🔔 Manejar interrupciones

🔄 Gestión del sistema

⚙️ Funciones del SO

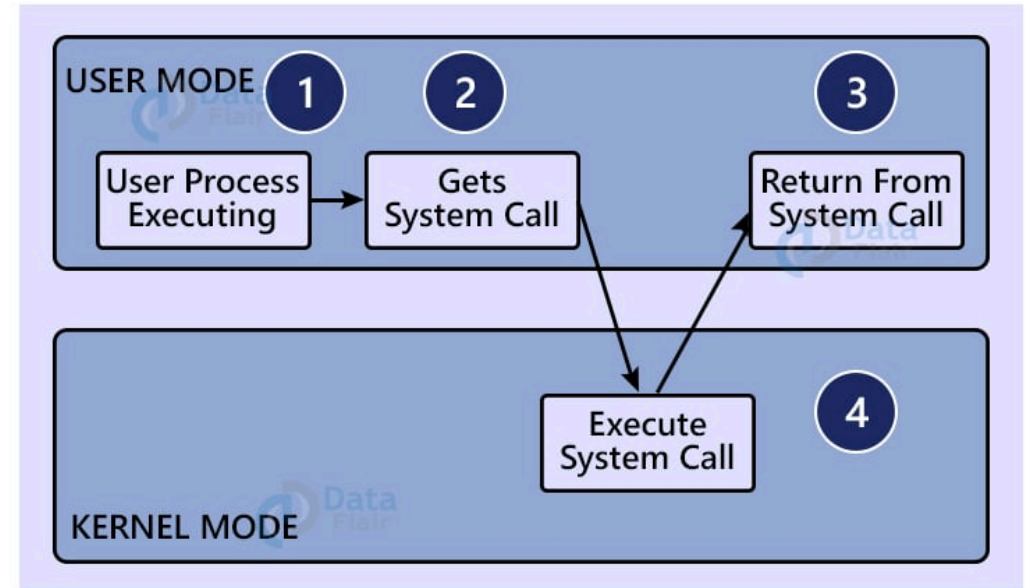
- Envía **comandos y datos** a dispositivos
- Conoce el **estado** de los dispositivos
- Detecta y gestiona **interrupciones**
- Controla y corrige **errores**



Mecanismo de ejecución

- 1 Preparación**
Programa prepara los parámetros necesarios
 - 2 Transición al modo kernel**
Ejecución de instrucción especial (INT 0x80, SYSCALL)
 - 3 Tratamiento por el kernel**
Verificación de parámetros y ejecución
 - 4 Retorno al espacio usuario**
Devolución del resultado y transferencia de control
-  Las llamadas al sistema son **costosas en tiempo** debido al cambio de contexto y transición entre modos de ejecución

WORKING OF A SYSTEM CALL



Relación con interrupciones

↔ Mecanismo similar

Las llamadas al sistema utilizan un mecanismo similar al de las interrupciones hardware, pero se activan mediante **instrucciones específicas** en lugar de señales externas.

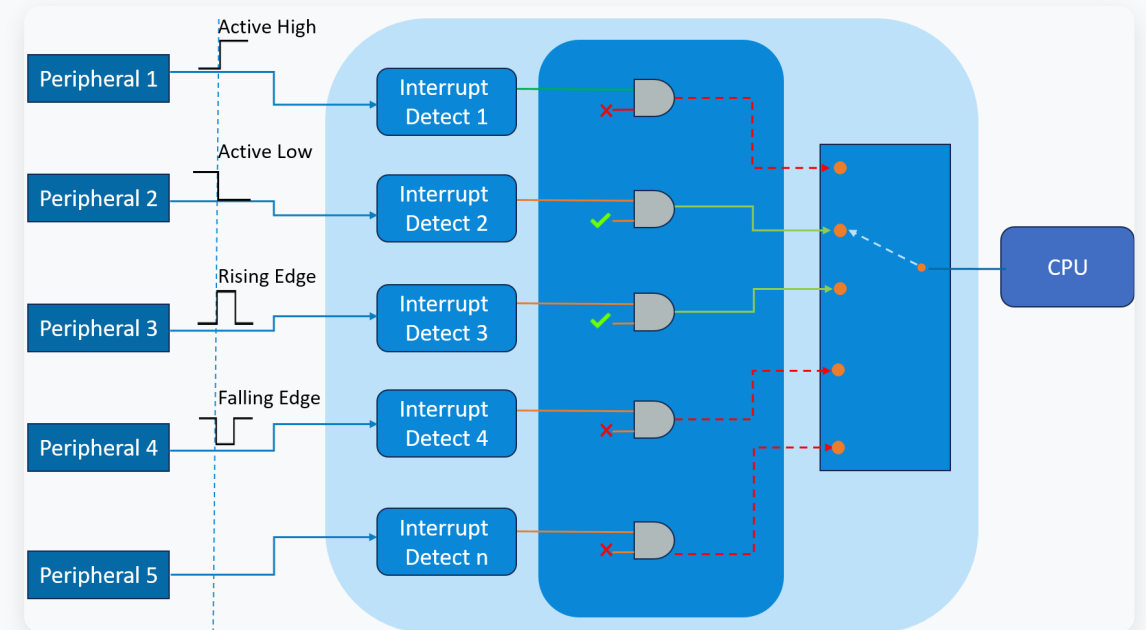
"Una interrupción provoca la detención momentánea de la ejecución del programa y la bifurcación a una posición de memoria donde comienza la rutina de tratamiento de la interrupción (RTI), que se encarga de realizar las operaciones de E/S y, posteriormente, retorna al punto donde se había interrumpido la ejecución del programa."

✓ Similitudes

- Cambio de contexto
- Transferencia de control al kernel
- Ejecución de rutinas específicas

⊗ Diferencias

- Activación: **interna** vs externa
- Propósito: **solicitado** vs asíncrono
- Frecuencia: predecible vs impredecible



Importancia en la arquitectura del sistema operativo



Puente fundamental entre la capa de aplicación y el núcleo del sistema operativo

◆ Jerarquía de niveles

4

Lenguajes de alto nivel

3

Sistema Operativo (contiene las llamadas al sistema)

2

Lenguaje máquina

1

Microarquitectura

0

Lógica digital

↗ Impacto del diseño

🔗 Eficiencia

🛡️ Seguridad

👤 Usabilidad

📦 Portabilidad

