

Unitat 1 · Sistemes Operatius (SO)

Estructura dels Sistemes Operatius

Jordi Mateo jordi.mateo@udl.cat

Escola Politècnica Superior (EPS) <https://www.eps.udl.cat/> · Departament d'Enginyeria Informàtica i Disseny Digital <https://deidd.udl.cat/>

Crides a sistema

Que són les crides a sistema (System Calls)?

Les **crides al sistema** ofereixen les funcions bàsiques per a poder utilitzar i interactuar amb els recursos del sistema de manera correcta i controlada.

Nº de crides a sistema

- Linux: aprox. 300
- FreeBSD: aprox. 500
- Windows: aprox. 2000

Usos comuns

- Crear, obrir, tancar i eliminar fitxers.
- Crear i gestionar nous processos.
- Crear i gestionar la xarxa.

Utilització

- Per accedir a les crides de sistema necessitem les **Llibreries del sistema**.
- Per exemple:
 - Unix: *stdio,stdlib,sys/shm,...*
 - Windows DLL: *system32.dll, ..*

Quines són les principals crides a sistema?

	Windows	Unix
Control de processos	CreateProcess()	fork()
	ExitProcess()	exit()
	WaitForSingleObject()	wait()
Manteniment & Informació	GetCurrentProcessID()	getpid()
	SetTimer()	alarm()
	Sleep()	sleep()
Comunicació	CreatePipe()	pipe()
Protecció	SetFileSecurity()	chmod()
	SetSecurityDescriptorGroup()	chown()
Manipulació de fitxers	CreateFile()	open()
	ReadFile()	read()
	WriteFile()	write()
	CloseHandle()	close()

Exemple amb la llibreria (*stdio.h*)

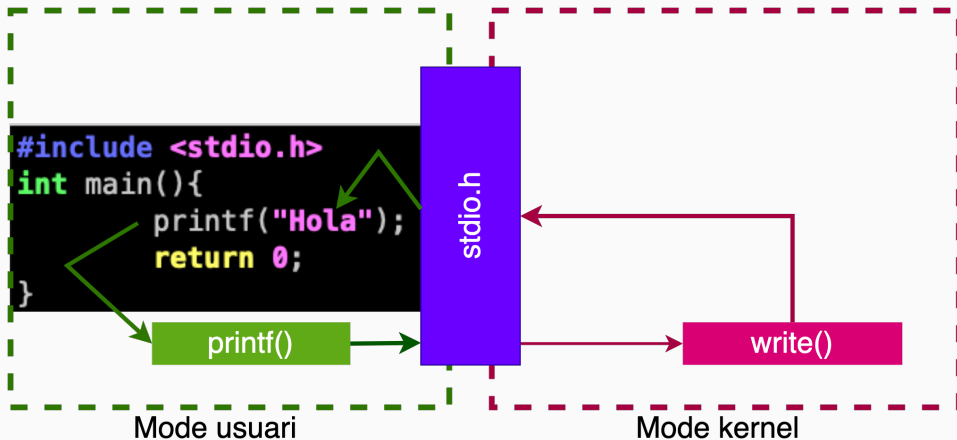


Figura 1: Esquema del funcionament intern de `printf`

Nucli

Què és el nucli d'un sistema operatiu?

El nucli del **SO** és la capa més crítica i conté les rutines de gestió del sistema relacionades amb els recursos físic. Es troba sempre carregat a la Memòria.

Funcionalitats del nucli

- El nucli té la capacitat d'assignar i desassignar la CPU als usuaris i processos per evitar que acaparin el recurs de forma ininterrompuda.

Què és el nucli d'un sistema operatiu?

El nucli del **SO** és la capa més crítica i conté les rutines de gestió del sistema relacionades amb els recursos físic. Es troba sempre carregat a la Memòria.

Funcionalitats del nucli

- El nucli té la capacitat d'assignar i desassignar la CPU als usuaris i processos per evitar que acaparin el recurs de forma ininterrompuda.
- El nucli evita que els usuaris i processos accedeixin a dades d'altres usuaris.

Què és el nucli d'un sistema operatiu?

El nucli del **SO** és la capa més crítica i conté les rutines de gestió del sistema relacionades amb els recursos físic. Es troba sempre carregat a la Memòria.

Funcionalitats del nucli

- El nucli té la capacitat d'assignar i desassignar la CPU als usuaris i processos per evitar que acaparin el recurs de forma ininterrompuda.
- El nucli evita que els usuaris i processos accedeixin a dades d'altres usuaris.
- El nucli evita els usuaris modifiquin el codi i les dades del nucli.

Què és el nucli d'un sistema operatiu?

El nucli del **SO** és la capa més crítica i conté les rutines de gestió del sistema relacionades amb els recursos físic. Es troba sempre carregat a la Memòria.

Funcionalitats del nucli

- El nucli té la capacitat d'assignar i desassignar la CPU als usuaris i processos per evitar que acaparin el recurs de forma ininterrompuda.
- El nucli evita que els usuaris i processos accedeixin a dades d'altres usuaris.
- El nucli evita els usuaris modifiquin el codi i les dades del nucli.
- El nucli evita que els usuaris realitzin E/S il·legals.

Qué és la dualitat?

Mode kernel

- El codi que s'executa en aquest mode té accés a qualsevol adreça de Memòria i a tots els recursos hardware.
- Si un programa falla en aquest mode, tot el sistema quedarà aturat.

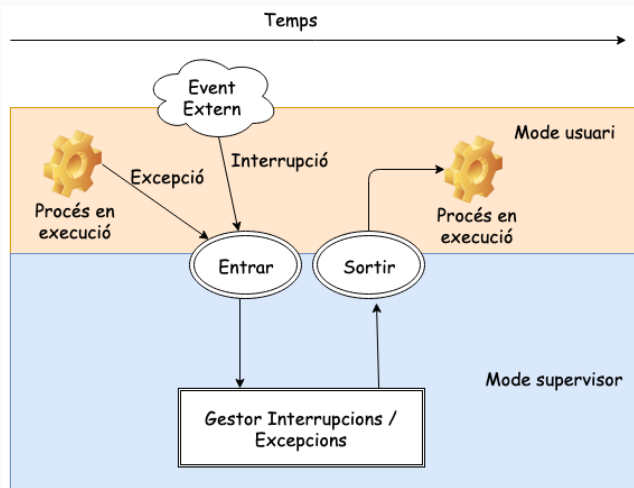
Mode usuari

- El codi no té accés directe a Memòria ni als recursos hardware.
- Si un programa falla en aquest mode, únicament atura el programa i no el sistema.

Observació

En mode usuari la CPU comprova cada instrucció per comprovar que el procés pot realitzar-la. En el mode kernel s'executen sense cap comprovació de protecció.

Com accedim al Kernel



TRAPS són mecanismes que permeten als processos d'aplicacions sol·licitar serveis del kernel. Un procés d'aplicació pot demanar funcions o operacions que només el kernel té permisos per realitzar.

Exemple de trap: `read(fd, buffer, nbytes)`

Espai d'usuari

1. Programa C:
 - Posa *fd*, *buffer*, *nbytes* als registres(*RDI*,*RSI*,*RDX*).
 - Crida a `read()`.
2. Llibreria de sistema:
 - Posa el codi de `read` a *RAX*.
 - Executa `syscall` (TRAP).

Transició a mode kernel

Espai del kernel

- Crida al planificador de tasques.
- Localitza el handler de `read`.
- Executa el handler de `read`:
 - Llegeix dades del dispositiu d'E/S.
 - Col·loca les dades al buffer d'usuari.
- Retorna a l'espai d'usuari.

Transició a mode usuari

Espai d'usuari

- **Llibreria de sistema:**
Retorna el resultat de la crida a `read()` al Programa C.
- **Programa:** Continua l'execució utilitzant les dades llegides. O bé, tracta l'error si n'hi ha (*errno*)

Què és una interrupció?

Les interrupcions són esdeveniments **hardware o software** asíncrons independents al procés que actualment s'està executant.

Gestió

1. Guardar el context del procés actual.
2. Canviar a mode kernel.
3. Determinar la causa de la interrupció.
4. Buscar la direcció de la RTI.
5. Cridar a la RTI.
6. Restaurar el context del procés.
7. Tornar a mode usuari

El seu tractament és **prioritari**.

Generalment són errors. Trencaments de la seqüència no previstos provocats per l'execució en curs de l'usuari (condició anormal mentre s'executa una instrucció.). El Sistema operatiu intenta tractar l'excepció si no pot solucionar el problema, s'enviarà un senyal al procés. El procés tracta el senyal executant el gestor per defecte o específic del procés per aquell senyal.

Faults

Un fault és un tipus d'excepció que s'informa abans de l'execució de la instrucció i que normalment es pot corregir. (per exemple, error de pàgina).

Traps

Una trap s'informa després de l'execució de la instrucció en què s'ha detectat l'excepció. (per exemple, trap de depuració).

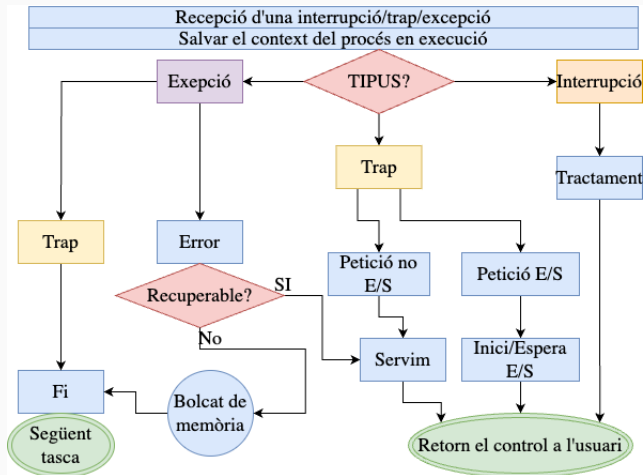
Aborts

Errors que no es poden corregir.

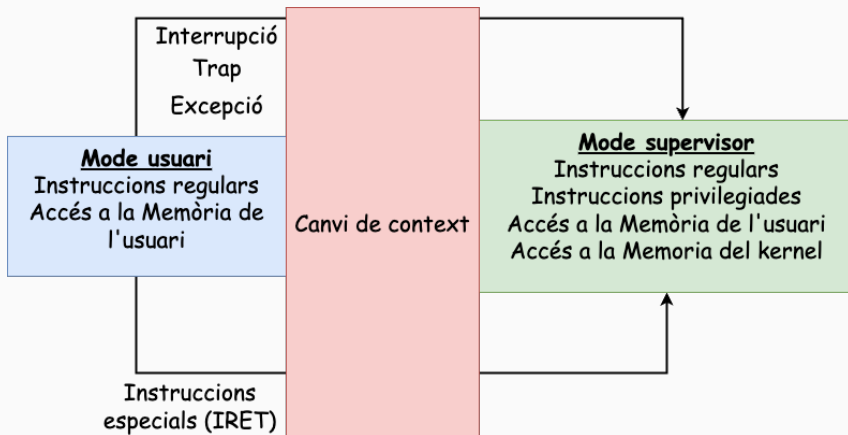
Exemples d'exceptions

Name	Type
Divide-by-zero	Fault
Debug	Fault/Trap
Non-maskable Interrupt	Interrupt
Breakpoint	Trap
Overflow	Trap
Bound Range Exceeded	Fault
Device not Available	Fault
Double Fault	Abort
Segment Not Present	Fault
Page Fault	Fault

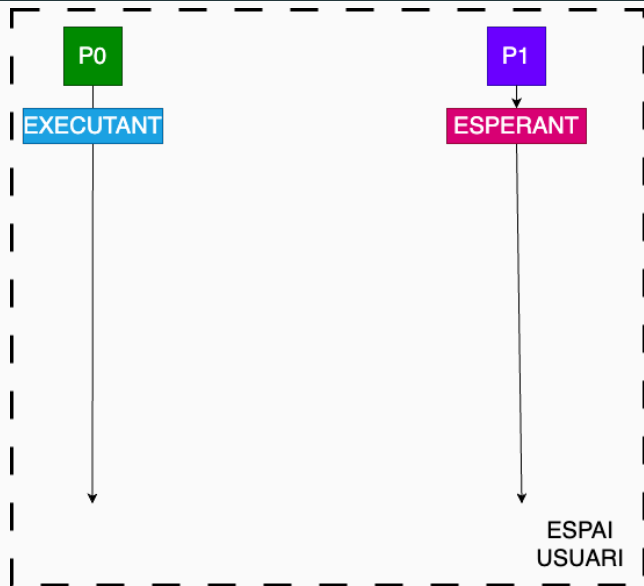
Procediment general



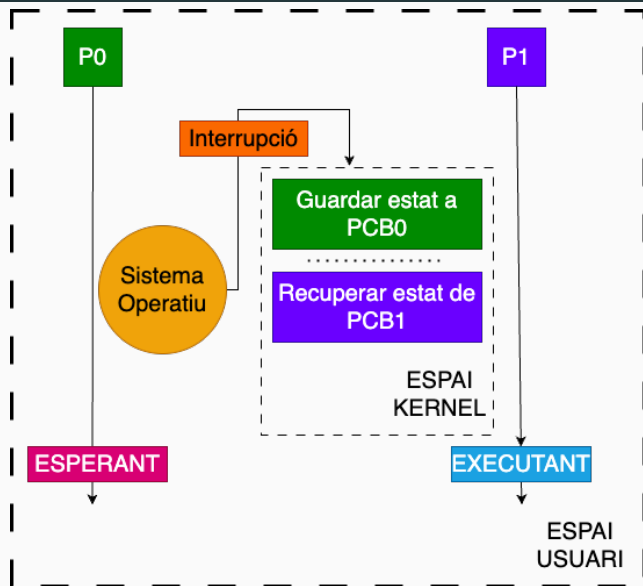
Com funciona la dualitat?



Canvi de context (I)



Canvi de context (II)



Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.
- Per accedir al nucli necessitem realitzar un **canvi de context**.

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.
- Per accedir al nucli necessitem realitzar un **canvi de context**.
- Les **crides a sistema** permeten utilitzar les rutines del nucli.

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.
- Per accedir al nucli necessitem realitzar un **canvi de context**.
- Les **crides a sistema** permeten utilitzar les rutines del nucli.
- **Instruccions privilegiades**. Totes les instruccions potencialment insegures estan prohibides quan s'executen en mode d'usuari.

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.
- Per accedir al nucli necessitem realitzar un **canvi de context**.
- Les **crides a sistema** permeten utilitzar les rutines del nucli.
- **Instruccions privilegiades**. Totes les instruccions potencialment insegures estan prohibides quan s'executen en mode d'usuari.
- **Protecció de memòria**. Tots els accessos a la memòria fora de la regió de memòria vàlida d'un procés estan prohibits quan s'executen en mode d'usuari. Transforma les referències lògiques (abstractes) en físiques (concretes).

Quines són les característiques de la dualitat

- El codi que pertany al nucli sempre s'executa amb tots els privilegis (**mode supervisor**).
- El SO minimitza el temps i el codi que s'executa aquest mode.
- Per accedir al nucli necessitem realitzar un **canvi de context**.
- Les **crides a sistema** permeten utilitzar les rutines del nucli.
- **Instruccions privilegiades**. Totes les instruccions potencialment insegures estan prohibides quan s'executen en mode d'usuari.
- **Protecció de memòria**. Tots els accessos a la memòria fora de la regió de memòria vàlida d'un procés estan prohibits quan s'executen en mode d'usuari. Transforma les referències lògiques (abstractes) en físiques (concretes).
- **Interrupció del temporitzador**. Independentment del que faci el procés. El nucli pot recuperar periòdicament el control del procés actual. Errors que no es poden corregir.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.
- Canvieu els modes de processador del nucli a l'usuari.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.
- Canvieu els modes de processador del nucli a l'usuari.
- Canvieu el voltatge i la freqüència del processador.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.
- Canvieu els modes de processador del nucli a l'usuari.
- Canvieu el voltatge i la freqüència del processador.
- Parar un processador.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.
- Canvieu els modes de processador del nucli a l'usuari.
- Canvieu el voltatge i la freqüència del processador.
- Parar un processador.
- Restableix un processador.

Exemples d'instruccions privilegiades

- Assignació d'adreces de memòria.
- Esborreu o invalideu la memòria cau de dades.
- Invalideu les entrades a les taules de pàgines.
- Carregueu i llegiu els registres del sistema.
- Canvieu els modes de processador del nucli a l'usuari.
- Canvieu el voltatge i la freqüència del processador.
- Parar un processador.
- Restableix un processador.
- Fer operacions d'E/S.

Estuctures dels sistemes operatius

Quina és la problemàtica?

Quina és la millor manera d'organitzar/separar totes les parts del sistema operatiu?

Reptes

- Com **organitzem** les parts?
- **Definició de polítiques**: Quines seran les accions a realitzar.
- Com **cooperen** les parts?
- **Mecanismes**: Com es duran a terme aquestes accions.

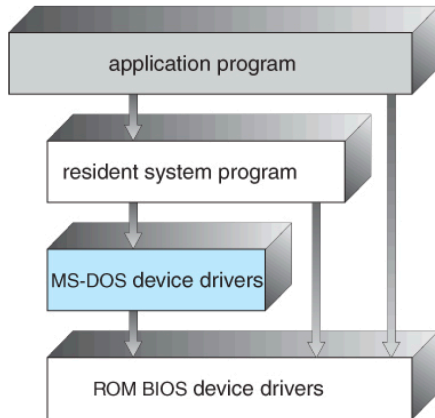
Classificació (Estructura Interna)

1. Monolítics
2. Capes
3. Micro-kernel
4. Híbrids
5. Màquines Virtuals

Estructura simple

Característiques

- Estructura no ben definida.
- El **SO** és un conjunt de procediments que es poden cridar sense cap limitació.
- Son sistemes complexos; difícil d'implementar i de depurar.
- No tenen mode **dual**.
- Exemple: MS-DOS



Estructura Monolítica

Característiques de l'estructura monolítica

Els serveis d'usuari i serveis del kernel s'implementen sota el mateix espai d'adreces.

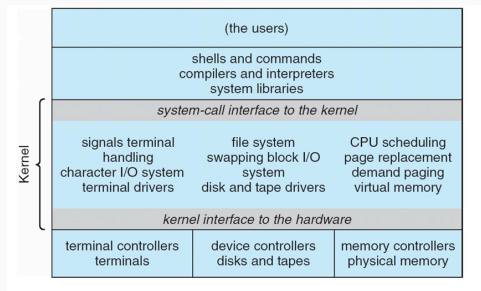


Figura 2: Esquema d'un kernel monolític

Pros de l'estructura monolítica

- Les funcionalitats (serveis) s'invoquen amb crides al sistema.
- Els controladors de dispositius es carreguen al nucli i passen a formar part del nucli.

Cons de l'estructura monolítica

- Difícil d'entendre, modificar i mantenir.
- Poc fiable (sense aïllament entre els mòduls del sistema)

Estructura per capes

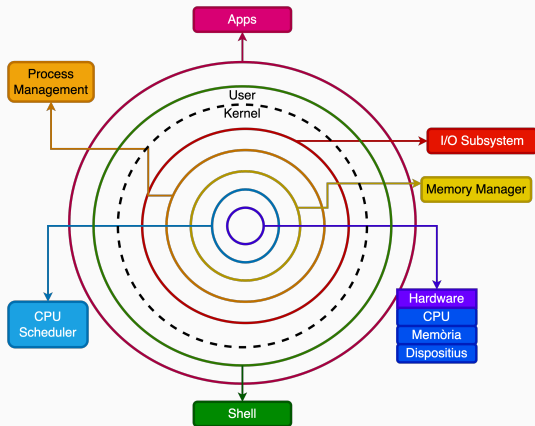


Figura 3: Esquema del kernel per capes

Pros de l'estructura per capes

- Independència entre les capes.
- Permet descriure el **SO** de forma clara.
- Simplicitat en la construcció i depuració.

Cons de l'estructura per capes

- Rendiment.
- És difícil definir les capes a causa de les limitacions per comunicar-se.

Estructura Microkernel

Característiques de l'estructura Microkernel

- Els serveis d'usuari i serveis del kernel s'implementi en diferents espais d'adreces.
- Comunicació entre els mòduls utilitza el pas de missatges.

Pros de l'estructura Microkernel

- El kernel té una mida més reduïda.
- Portable, segur, fiable i extensible.

Cons de l'estructura Microkernel

- Reducció de la velocitat d'execució i del rendiment.

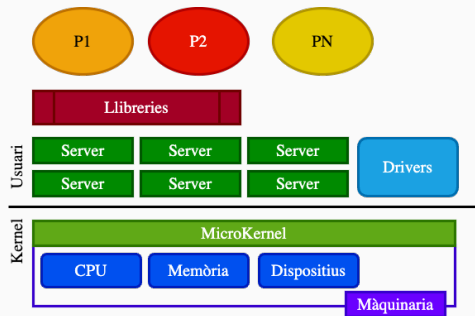
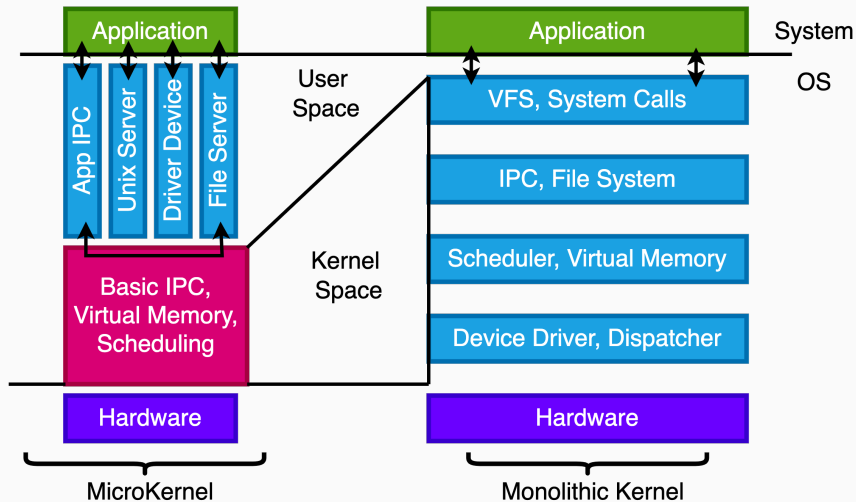
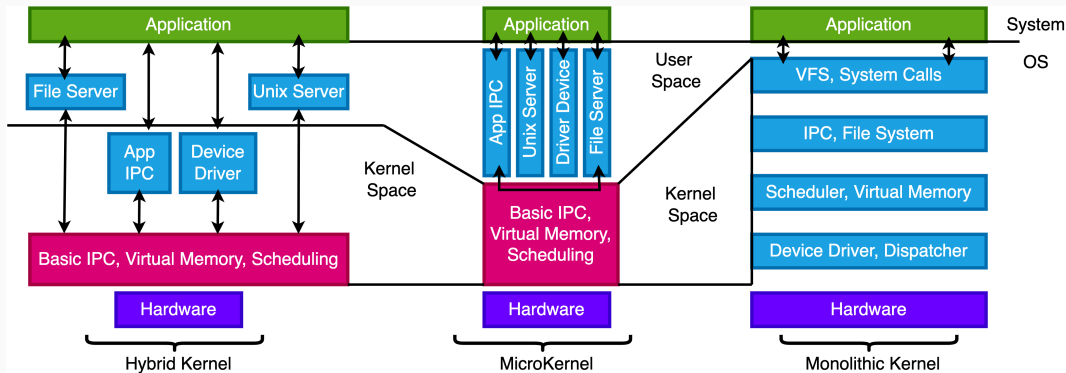


Figura 4: Esquema d'un microkernel

Comparativa Microkernel vs Monolítico



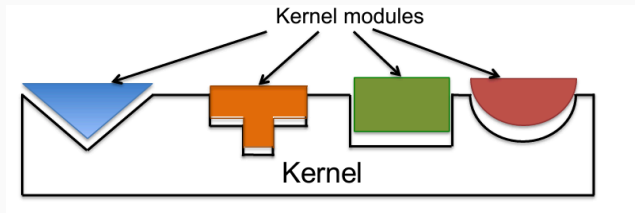
Comparativa Microkernel vs Monolítico vs Híbridos



Definició

El serveis (*core*) estan integrats al kernel, la resta es poden carregar i descarregar de forma dinàmica.

- No cal reiniciar per afegir nous mòduls.
- No cal implementar mecanismes de pas de missatges com en els microkernels.
- Qualsevol mòdul pot comunicar-se amb qualsevol altre.

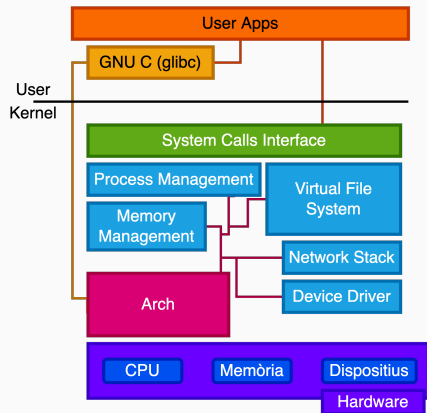


Exemple: Linux

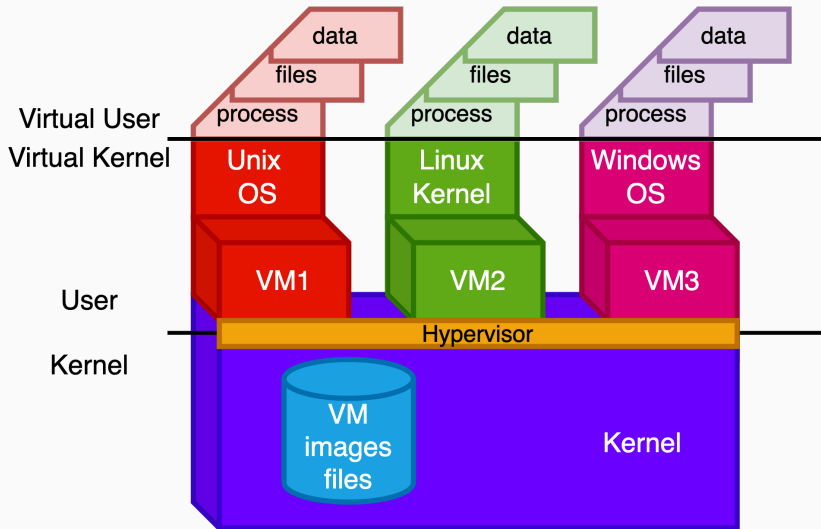
Definició

El nucli Linux és un dels projectes de codi obert més grans del món, amb milers de desenvolupadors que aporten codi i milions de línies de codi canviats per a cada versió.

- Arquitectura **Monolítica** híbrida basada en mòduls.
 - Enllaçat dinàmic.
 - Mòduls apilables.
- Disseny orientat a objectes.
- Suport per a múltiples fils d'execució.
- Suport per processament múltiple simètric.
- Abstracció hardware.

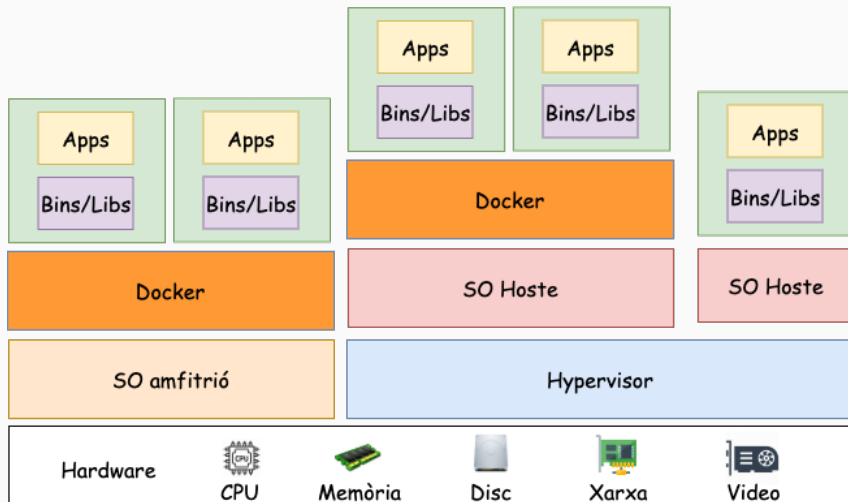


Màquines Virtuals



JVM (Java Virtual Machine): Ens permet executar codi Java en qualsevol plataforma.

Contenidors



Arquitectura **minimalista** que intenta proporcionar als programes un control directe sobre el maquinari. A diferència dels microkernels o dels kernels monolítics, els exokernels no intenten abstraure gairebé res del maquinari, sinó que ofereixen primitives de baix nivell per accedir directament als recursos de la màquina. La gestió de recursos, com la memòria i la CPU, queda delegada als programes d'usuari o a les seves llibreries.

Imaginem que particionem una màquina en diferents màquines virtuals amb un subconjunt de recursos. Cada màquina virtual té un exokernel que li permet accedir directament als recursos de la màquina física. La idea és reduir el overhead del sistema separant la multiprogramació de les funcions del sistema operatiu en l'espai usuari.

Els unikernels són una forma de sistema operatiu que empaqueta tot el codi necessari per a una aplicació en un sol paquet. Aquest paquet s'executa directament sobre una màquina virtual o un hipervisor, sense cap sistema operatiu subministrat. Això permet als unikernels ser molt petits i molt eficients, ja que no hi ha cap codi innecessari.

S'inspiren en els contenidors, però en lloc d'executar-se en un sistema operatiu complet, s'executen directament sobre el hardware. Això els fa molt més petits i molt més eficients que els contenidors, ja que no hi ha cap sistema operatiu subministrat.

S'inspiren en els exokernels, però en lloc de proporcionar un conjunt de primitives de baix nivell per accedir directament als recursos de la màquina, proporcionen un conjunt de primitives de baix nivell per accedir directament als recursos de l'aplicació.

PREGUNTES?

Materials del curs

- Organització — OS-GEI-IGUALADA-2425
- Materials — Materials del curs
- Laboratoris — Laboratoris
- Recursos — Campus Virtual

TAKE HOME MESSAGE: La dualitat i el disseny d'estructures eficients són crucials per a una gestió segura i òptima dels recursos i processos, garantint alhora un funcionament estable i eficaç dels dispositius informàtics.



Figura 5: Això és tot per avui