



# UD1. CONCEPTO DE SISTEMA OPERATIVO. ELEMENTOS Y ESTRUCTURA DEL SISTEMA OPERATIVO.

---

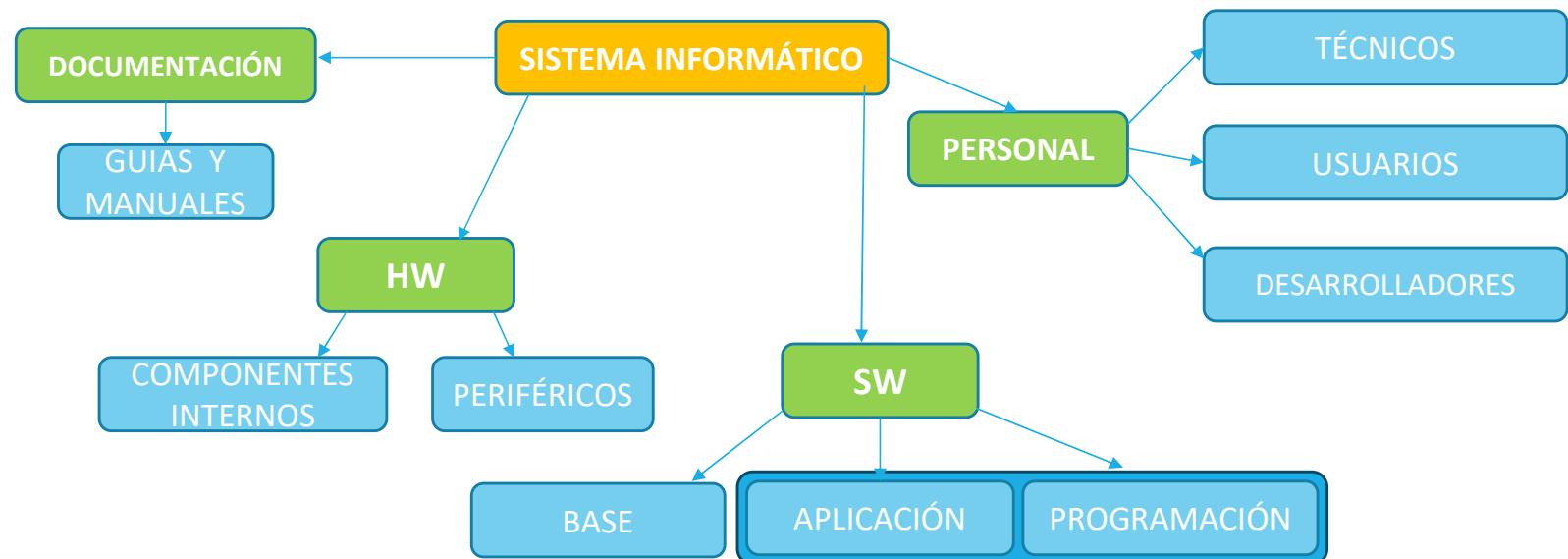
IES Comercio

# PARTES FUNDAMENTALES PARA EL FUNCIONAMIENTO



# ¿QUÉ ES UN SISTEMA INFORMÁTICO?

Un sistema informático es un conjunto de partes interrelacionadas que permiten **ALMACENAR** y **PROCESAR** la información.



# SW DEL ORDENADOR

***SOFTWARE:** conjunto de los **programas de cómputo, procedimientos, reglas, documentación y datos** asociados que forman parte de las operaciones de un sistema de computación.* (IEEE 729)

- **Tipos de software:**
  - Según el tipo de trabajo
  - Según método de distribución

# SW SEGÚN EL TIPO DE TRABAJO

✓ De sistema/base (Permiten que el HW funcione)



- S.O.
- Drivers
- SW Utilidad (Htas diagnostico y optimización)

✓ De programación. (IDE / GUI)



- Editores, Compiladores, Interpretes...
- IDE

✓ De aplicación. (Tareas específicas)



- Control y Aplicación Industrial
- Ofimáticas
- SGBD
- Multimedia
- Diseño gráfico
- SW Dedicado (medicina, educación, empresarial...)

# SW SEGÚN EL MÉTODO DE DISTRIBUCIÓN

## ❖ SW PROPIETARIO

- ✓ **Freeware.** (Sin cargo, mantiene los dchos autor)  
(≠ SW libre)
- ✓ **Shareware.** (Evaluación)
- ✓ **Demoware** (No completa)
- ✓ **Trial** (Completa pero tiempo limitado)
- ✓ **Adware** (Con publicidad)
- ✓ **COMERCIAL**
  - ✓ **RETAIL** (Al por menor)
  - ✓ **OEM** (Ligada al equipo)
    - OEM por lotes (preinstalada)

## ❖ SW LIBRE

# LICENCIAS

Una licencia de software es una forma de **contrato** mediante el que el titular de los derechos de autor establece las **condiciones** y términos bajo los que el usuario puede utilizar un determinado programa informático y los **derechos** que adquiere el usuario al adquirir la licencia.



Usuario



# LICENCIAS

- **Software privativo:** reserva la mayoría de los derechos de modificación, duplicación y redistribución para el titular de los derechos de propiedad intelectual (el programador o, en ocasiones, el distribuidor). Relacionadas con el concepto de Copyright.
  - Ejemplo: EULA



- **Software libre:** puede ser usado copiado, estudiado, modificado y redistribuido libremente. El término está relacionado con las libertades permitidas a los usuarios, no con el **precio** del mismo. Relacionadas con el concepto de Copyleft. Existen distintas tipos:
  - Ejemplos: GNU GPL, Creative Commons, Copy Left, BSD.

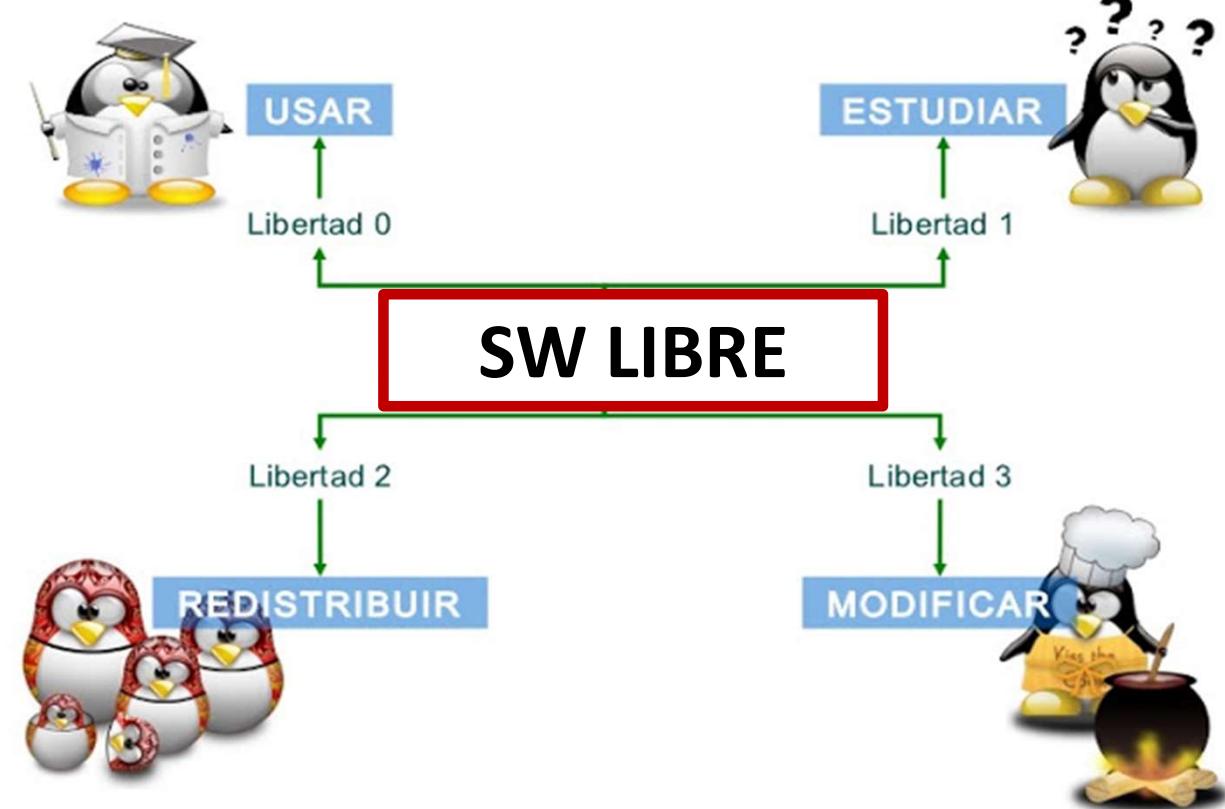


# SOFTWARE LIBRE – FREE SOFTWARE

|  |                       |
|--|-----------------------|
|  | Libertad 0: Usar      |
|  | Libertad 1: Analizar  |
|  | Libertad 2: Compartir |
|  | Libertad 3: Mejorar   |

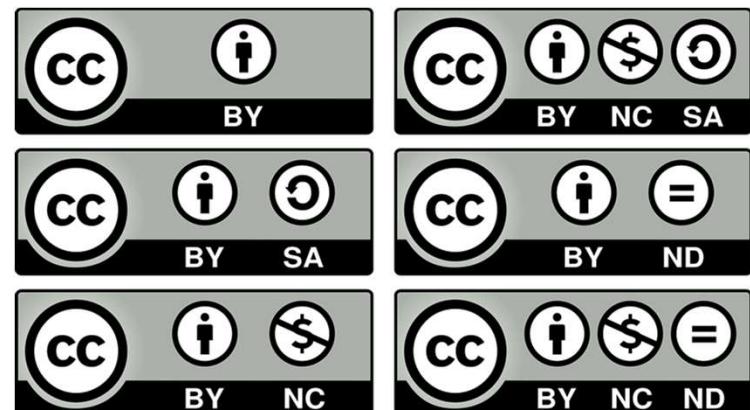


**FREE SOFTWARE FOUNDATION**



# LICENCIAS

- **Creative commons.** Establecen diferentes condiciones que se pueden combinar.
  - Atribución (BY): para usar una obra en cualquier tipo de medio es imprescindible citar al autor de forma explícita.
  - Compartirlgual (SA): se puede usar una obra para crear otra, siempre y cuando esta se publique con la misma licencia que la obra original.
  - NoDerivadas (ND): puedes usar una obra para cualquier cosa, siempre y cuando no la modifiques. Un ejemplo sería coger una foto de un cielo y editarla para añadir un avión pasando por encima, o modificar la velocidad de un clip de audio.
  - NoComercial (NC): puedes usar una obra para generar otra obra, siempre y cuando no vayas a ganar dinero con ella. Es muy común en las fotos de Flickr.
- **BSD.** Licencia libre flexible respecto a la redistribución, ya que las modificaciones se pueden redistribuir de forma libre o privativa.



FreeBSD®

# FREEWARE VS SOFTWARE LIBRE

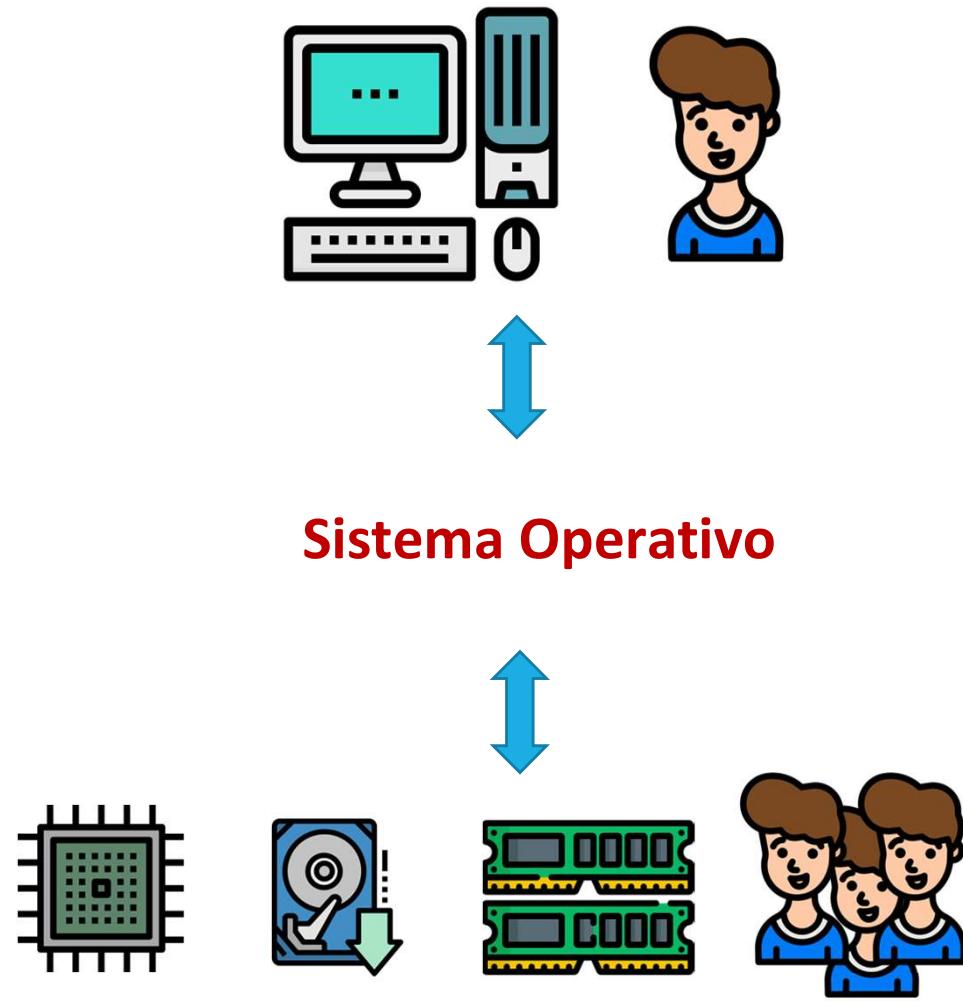
## EJERO1

# ¿Qué es el SO?

El Sistema Operativo es un asignador de recursos

Dos objetivos principales:

- **GESTOR DE RECURSOS:** Maneja el HW de forma efectiva y eficiente
- **MÁQUINA SIMBÓLICA:** Facilita el uso del ordenador al usuario

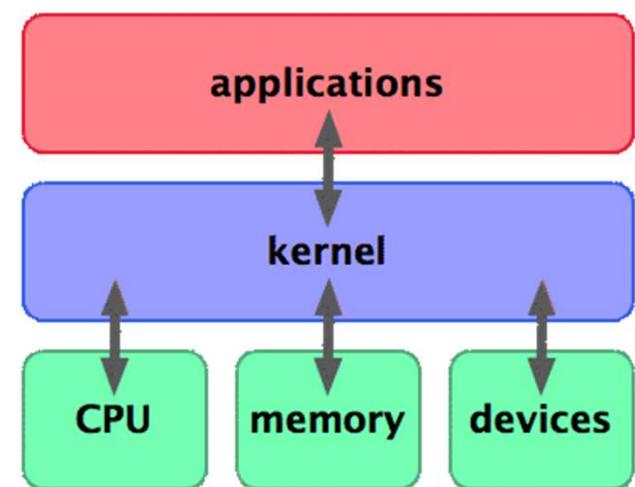


# ELEMENTOS DE UN SO

NÚCLEO o KERNEL

UTILIDADES DE BAJO NIVEL

INTERFAZ DE USUARIO



# INTERFACES



# FUNCIONAMIENTO DEL SO

## NÚCLEO: MODO PRIVILEGIADO / KERNEL

- ACCESO A TODO EL HW
- PUEDE EJECUTAR CUALQUIER INSTRUCCIÓN QUE EL EQUIPO SEA CAPAZ



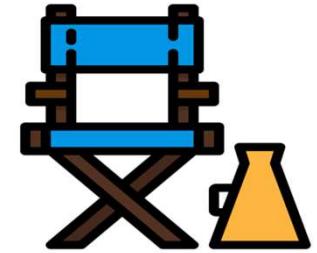
## UTILIDADES DE BAJO NIVEL: MODO USUARIO

- ACCESO Y EJECUCIÓN LIMITADOS

# FUNCIONES

Las más importantes:

- Administrador del procesador
- Administrador de la memoria
- Gestión de los dispositivos de E/S (periféricos)
- Administración del sistema de archivos
- Detección y tratamiento de los errores (control de errores)
- Seguridad y protección del sistema (control de seguridad)
- Control de redes



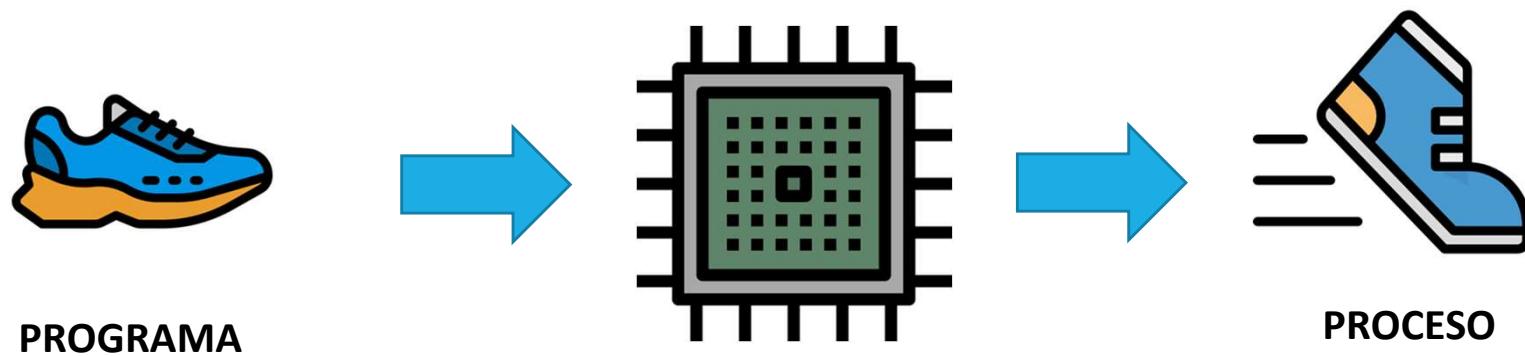
Funcionamiento

MODO KERNEL

MODO USUARIO

## FUNCIONES. ADMINISTRADOR DEL PROCESADOR

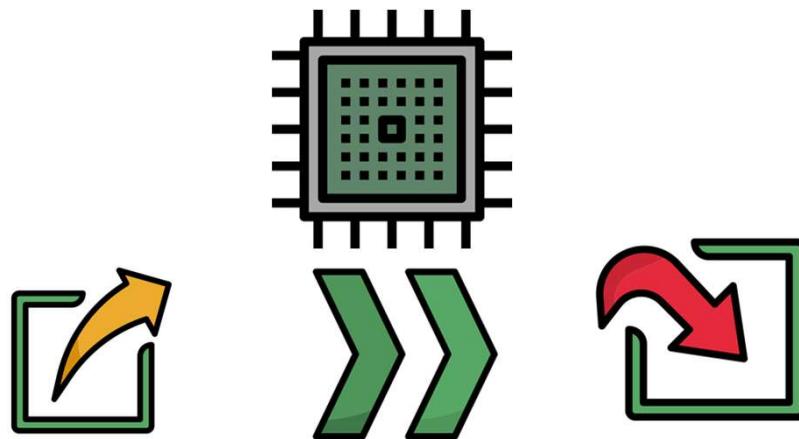
### PROCESO: programa en ejecución



## FUNCIONES. ADMINISTRADOR DEL PROCESADOR

El sistema operativo se encarga de compartir la CPU entre los distintos procesos con el propósito de ser eficaz, dar buen tiempo de respuesta y un alto rendimiento.

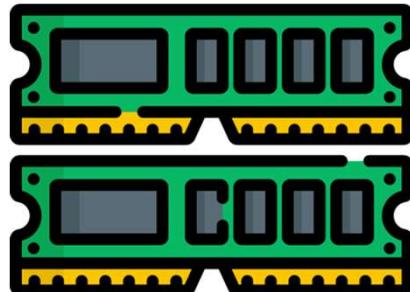
Para ello se utilizan **algoritmos de planificación**



# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

El administrador de memoria lleva en un **registro** las partes de memoria que se están utilizando y las que no. Así, **reservará** espacio de memoria para los nuevos procesos y **liberará** el espacio de los procesos que hayan finalizado.

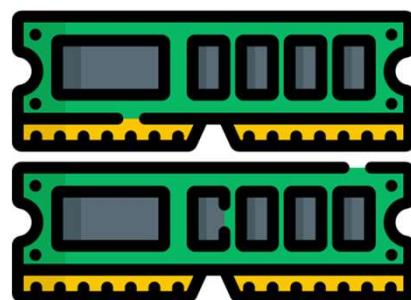


Memoria

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

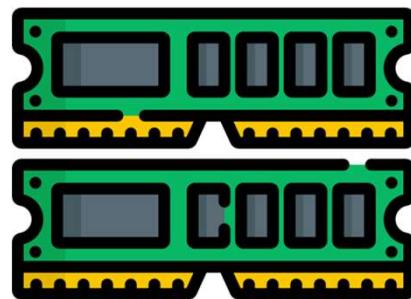
El administrador de memoria lleva en un registro las partes de memoria que se están utilizando y las que no. Así, reservará espacio de memoria para los nuevos procesos y liberará el espacio de los procesos que hayan finalizado.



# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

El administrador de memoria lleva en un registro las partes de memoria que se están utilizando y las que no. Así, reservará espacio de memoria para los nuevos procesos y liberará el espacio de los procesos que hayan finalizado.

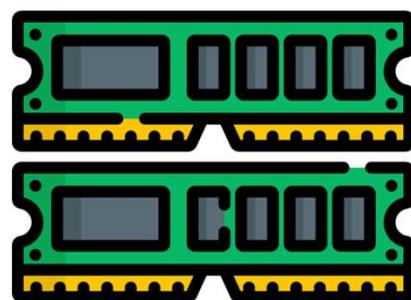


Memori  
a

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

El administrador de memoria lleva en un registro las partes de memoria que se están utilizando y las que no. Así, reservará espacio de memoria para los nuevos procesos y liberará el espacio de los procesos que hayan finalizado.

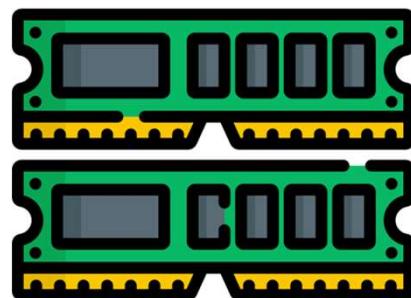


Memori  
a

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

El administrador de memoria lleva en un registro las partes de memoria que se están utilizando y las que no. Así, reservará espacio de memoria para los nuevos procesos y liberará el espacio de los procesos que hayan finalizado.

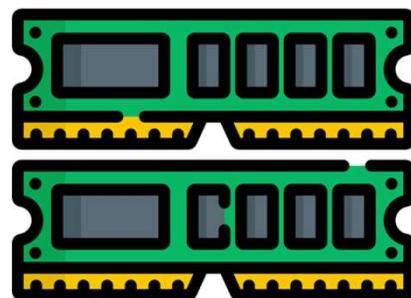


Memori  
a

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria no es capaz de contener todos los programas, datos y procesos que se realizan en un momento dado.

El administrador de memoria lleva en un registro las partes de memoria que se están utilizando y las que no. Así, reservará espacio de memoria para los nuevos procesos y liberará el espacio de los procesos que hayan finalizado.



Memori  
a

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

El objetivo es evitar la fragmentación o desaprovechamiento de memoria.

Para ello existen diferentes técnicas:

- **Segmentación.** Cada partición de memoria tiene un tamaño variable denominado segmento.
- **Paginación.** Cada partición de memoria tiene un tamaño fijo denominado marco de página.
- **Sistemas combinados.** Combinación de ambos.
- **Memoria Virtual.** Se utiliza parte del disco duro como memoria principal. (Swapping)



Memori  
a

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La fragmentación puede ser de dos tipos:

- **Fragmentación interna:** es la memoria que se encuentra **dentro de una partición** y no está siendo utilizada. Sin embargo, la partición está asignada a un proceso.
- **Fragmentación externa:** es la memoria que no se puede asignar por tener una capacidad **insuficiente para crear** una nueva partición. Estas zonas de memoria no se pueden asignar a ningún proceso y por lo tanto no se pueden utilizar.



Memori  
a

## DESFRAGMENTACIÓN

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 1 |   | 3 | 1 | 2 | 3 | 5 | 1 | 1 |
| 2 | 3 | 8 | 4 |   |   | 3 | 4 |   |
| 2 |   | 5 | 4 |   | 4 | 6 |   | 6 |
|   | 5 |   | 2 | 4 |   | 3 |   | 2 |
| 9 | 7 |   | 6 | 4 | 6 |   | 7 | 1 |
| 5 |   | 3 |   | 2 |   | 1 |   |   |
|   |   |   |   |   |   |   |   |   |

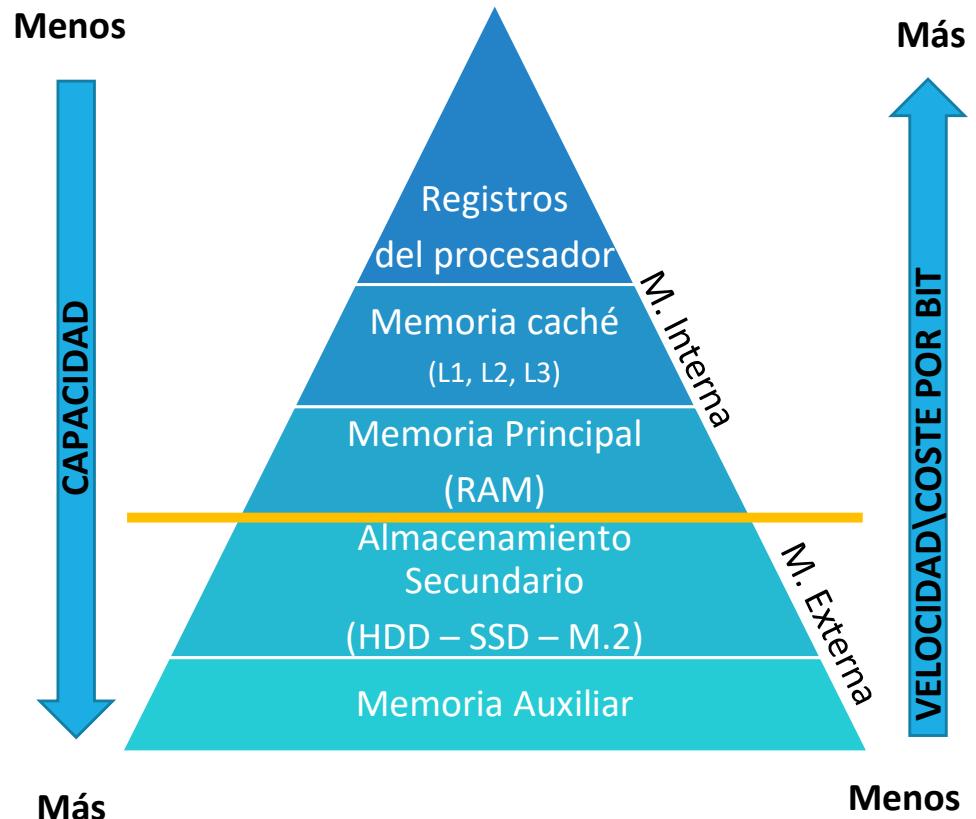
Cada color representa un archivo, y en un principio están todos fragmentados (esparcidos por todo el disco). Luego el proceso de desfragmentación va situando cada parte de cada archivo de tal manera que cada una quede en un área contigua a la anterior, evitando así que haya espacios entre ellos y reduciendo por tanto el tiempo de acceso a los archivos.

# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

La memoria se caracteriza por tres variables:

- Capacidad de almacenamiento
- Velocidad de acceso (lectura/escritura)
- **Tiempo de ciclo.** Tiempo que tarda entre la aceptación de una operación de lectura/escritura y la disponibilidad para aceptar la siguiente

Un proceso necesita disponer de un área de memoria donde colocar el código, los datos y la zona de trabajo o pila.



# FUNCIONES. ADMINISTRADOR DE LA MEMORIA

## Problema - ¿Cabe en memoria un programa con estos datos?

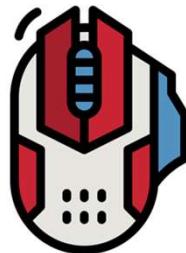
Tenemos una memoria de 64KB dividida en Marcos de 4KB y un programa con el siguiente tamaño:

- Código: 32.768 bytes
- Datos: 16.386 bytes
- Pila: 15.870 bytes

¿Y si el tamaño de la página fuera de 512 bytes?

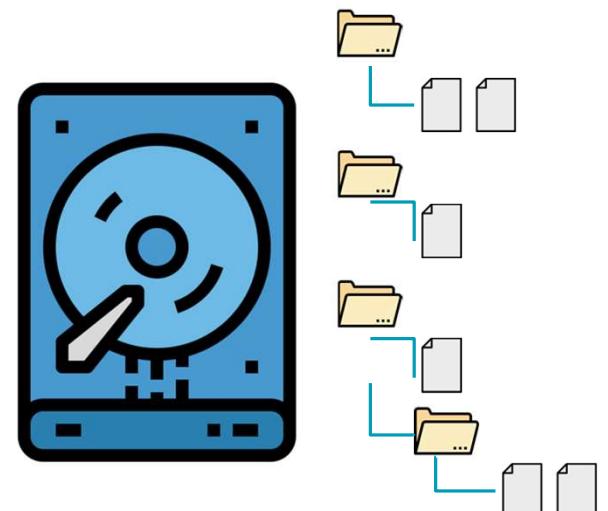
# FUNCIONES. ADMINISTRADOR DE E/S (PERIFÉRICOS)

- El S.O. lleva el control de todos los dispositivos de E/S
- Cada dispositivo de E/S tiene características distintas
- Solución adecuada en función del dispositivo
- Uso de **drivers**



# FUNCIONES. ADMINISTRACIÓN DEL SISTEMA DE ARCHIVOS

- Es necesario **organizar** la información para facilitar el acceso por parte del usuario
  - Se usa una **estructura jerárquica** de archivos y carpetas
- El sistema de archivos (**File System**) encargado de gestionar de que forma se guarda la información asociada a un archivo
- El sistema de archivos debe permitir al usuario realizar una serie de **operaciones** con los archivos y carpetas (**crear, borrar, mover, etc.**).



# FUNCIONES. CONTROL DE ERRORES

- El sistema operativo **supervisa** todas las operaciones que realizan los programas. Ello le permite detectar si se producen errores, por ejemplo, cuando no encuentra un dato, no reconoce una instrucción, se produce algún error en la ejecución, descubre algún defecto físico del hardware, etc.
- Cuando el sistema operativo diagnostica un error, activa unos programas para detectar su **causa** y si es posible **subsanarlo**, y si no, para **cancelar** el programa e informar al usuario sobre las características del error encontrado, si procede.



# FUNCIONES. CONTROL DE SEGURIDAD

## Dos mecanismos principales

- **Autenticación.** Garantiza la identidad de los usuarios, es decir, determina si un usuario es quien dice ser.
- **Privilegios.** Especifica qué recursos puede utilizar cada usuario, es decir, qué puede hacer con cada uno de ellos.



<https://www.elotrolado.net/noticias/scene/root-keys-ps5>

# EL ADMINISTRADOR DEL SISTEMA

El administrador de sistemas es la persona encargada del **buen funcionamiento** de un ordenador, sobre todo de su sistema operativo. Esto incluye tanto su **puesta en marcha** como su posterior **mantenimiento**.



# EL ADMINISTRADOR DEL SISTEMA

Resolver problemas que no sabe resolver



# TAREAS DEL ADMINISTRADOR DEL SISTEMA

- Añadir y eliminar **usuarios**, y asegurar así el correcto acceso a los recursos compartidos por éstos.
- Añadir y eliminar componentes físicos (**hardware**).
- Realizar periódicamente **copias de seguridad** de la información.
- **Instalar** aplicaciones de usuario.
- **Monitorizar** el sistema operativo, para poder ajustar los consumos de recursos, sobre todo los de proceso del sistema y memoria.
- Mantener la **documentación** del sistema.

## TAREAS DEL ADMINISTRADOR DEL SISTEMA

- Controlar el mantenimiento de la **seguridad**. Tanto conservando la integridad de los SO al chequearlos periódicamente, como ocupándose de la prevención y limpieza de virus informáticos e impidiendo el acceso a los no autorizados.
- Llevar a cabo el mantenimiento general de la **red**:
  - Mantener el SW que permite a los equipos conectarse en red.
  - Configurar y conservar conexiones que se realicen a otras redes, bien sean locales o externas.
  - Instalar, mantener y configurar impresoras conectadas en red.

# TAREAS DEL ADMINISTRADOR DEL SISTEMA

- Dar soporte a los usuarios.
- Resolver problemas.
- Velar por la integridad del sistema.

# ÉTICA DEL ADMINISTRADOR DEL SISTEMA

Hay una serie de normas que un buen administrador de sistemas debería seguir en su trabajo:

- Ser **profesional**.
- **Integridad** personal: ser capaz de admitir las propias limitaciones y/o errores.
- **Privacidad**: acceder a información privilegiada sólo en caso necesario y manteniendo siempre la confidencialidad.
- Estar en contacto con **leyes y políticas de uso**, manteniéndose siempre informado.
- Estar en **comunicación** con los usuarios, informándoles de los cambios del sistema que los afecten.
- Mantenerse **al día**.
- **Compartir** los conocimientos con los demás.
- Ser consciente de la importancia de la responsabilidad ética y social, predicando con el **ejemplo**.

# INTERFACES

Dos modos:

```
~/Documents/linux-commands via 🐧 v3.9.6
❯ ls -lah
Permissions Size User Date Modified Name
drwxr-xr-x  - daniel 8 ago 15:11 .
drwxr-xr-x  - daniel 8 ago 00:27 ..
drwxr-xr-x  - daniel 8 ago 00:34 commands
drwxr-xr-x  - daniel 7 ago 00:45 dir1
drwxr-xr-x  - daniel 7 ago 00:45 dir2
drwxr-xr-x  - daniel 8 ago 00:10 dir_to_copy
drwxr-xr-x  - daniel 8 ago 00:12 new_dir
.rw-r--r--  0 daniel 8 ago 00:38 BestMoviesOfAllTime
.rw-r--r--  0 daniel 7 ago 00:44 binarysearch.py
.rw-r--r--  0 daniel 7 ago 00:43 dummyfile1.txt
.rw-r--r--  0 daniel 8 ago 00:18 file_to_delete.txt
.rw-r--r--  0 daniel 7 ago 00:44 get_keys.py
.rw-r--r--  0 daniel 7 ago 00:44 github_automation.py
.rw-r--r--  0 daniel 7 ago 00:44 important_file.txt
.rw-r--r--  0 daniel 8 ago 00:04 new_file.txt
.rw-r--r--  0 daniel 8 ago 16:30 old_file
```

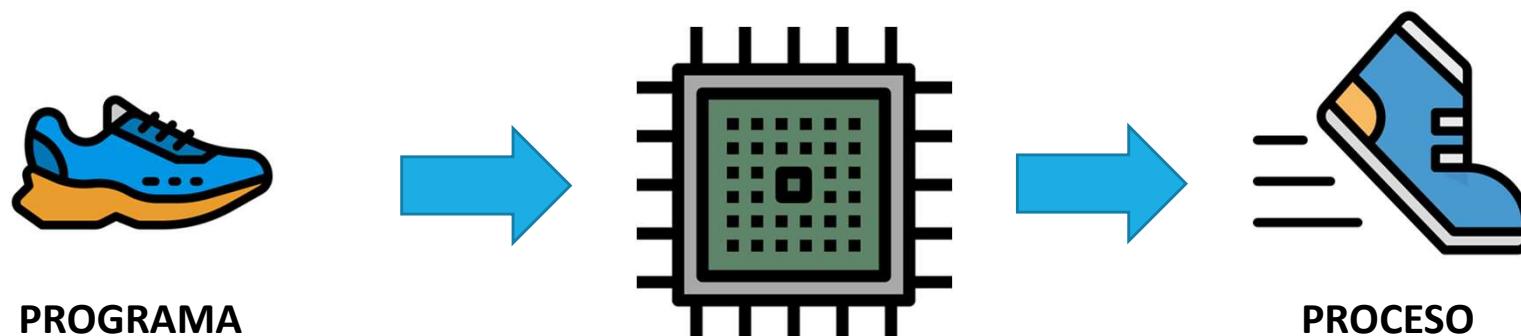
Línea de comandos - CLI



Interfaz gráfica - GUI

# PROCESOS

**PROCESO:** programa en ejecución



CADA PROCESO POSEE UN IDENTIFICADOR ÚNICO LLAMADO **ID**

# PROCESOS

| Nombre del proceso            | Usuario | % CPU | ID   | Memoria  | Lectura total | Escritura tot |
|-------------------------------|---------|-------|------|----------|---------------|---------------|
| at-spi2-registryd             | jaa     | 0,00  | 1843 | 696,3 kB | N/D           | N/L           |
| at-spi-bus-launcher           | jaa     | 0,00  | 1597 | 774,1 kB | N/D           | N/L           |
| dbus-daemon                   | jaa     | 0,11  | 1466 | 2,4 MB   | N/D           | N/L           |
| dbus-daemon                   | jaa     | 0,00  | 1613 | 495,6 kB | N/D           | N/L           |
| dconf-service                 | jaa     | 0,00  | 1808 | 561,2 kB | 77,8 kB       | 24,6 kB       |
| evolution-addressbook-factory | jaa     | 0,00  | 1814 | 3,5 MB   | 2,0 MB        | 36,9 kB       |
| evolution-alarm-notify        | jaa     | 0,00  | 1941 | 15,3 MB  | 1,4 MB        | N/L           |
| evolution-calendar-factory    | jaa     | 0,00  | 1800 | 3,9 MB   | 5,1 MB        | N/L           |
| evolution-source-registry     | jaa     | 0,00  | 1783 | 3,8 MB   | 3,3 MB        | N/L           |
| gdm-wayland-session           | jaa     | 0,00  | 1498 | 491,5 kB | N/D           | N/L           |
| gjs                           | jaa     | 0,00  | 1841 | 4,9 MB   | N/D           | N/L           |
| gjs                           | jaa     | 0,00  | 2088 | 5,0 MB   | N/D           | N/L           |
| gjs                           | jaa     | 0,34  | 2121 | 16,5 MB  | 2,0 MB        | N/L           |
| gnome-keyring-daemon          | jaa     | 0,00  | 1468 | 954,4 kB | 520,2 kB      | 4,1 kB        |
| gnome-session-binary          | jaa     | 0,00  | 1505 | 1,7 MB   | 2,2 MB        | N/L           |
| gnome-session-binary          | jaa     | 0,00  | 1578 | 3,1 MB   | 8,6 MB        | 4,1 kB        |
| gnome-session-ctl             | jaa     | 0,00  | 1565 | 442,4 kB | 20,5 kB       | N/L           |
| gnome-shell                   | jaa     | 8,47  | 1601 | 201,6 MB | 17,5 MB       | 49,2 kB       |
| gnome-shell-calendar-server   | jaa     | 0,00  | 1773 | 2,6 MB   | 6,0 MB        | N/L           |
| gnome-system-monitor          | jaa     | 4,74  | 3338 | 18,0 MB  | 151,6 kB      | N/L           |
| goa-daemon                    | jaa     | 0,00  | 1630 | 7,3 MB   | 4,1 kB        | N/L           |

# PROCESOS – MODO DE EJECUCIÓN (i)

## Desde el entorno de texto (línea de comandos)

- Ejecución en primer plano (foreground)
  1. Espera a que el usuario introduzca el comando
  2. Ejecuta el comando, y **cuando termina** su ejecución, vuelve al punto 1.
- Ejecución en segundo plano (background)
  1. Espera a que el usuario introduzca el comando
  2. Ejecuta el comando, y **cuando sin esperar a que termine** su ejecución, vuelve al punto 1.

# PROCESOS – MODO DE EJECUCIÓN (ii)

## Desde el entorno gráfico:

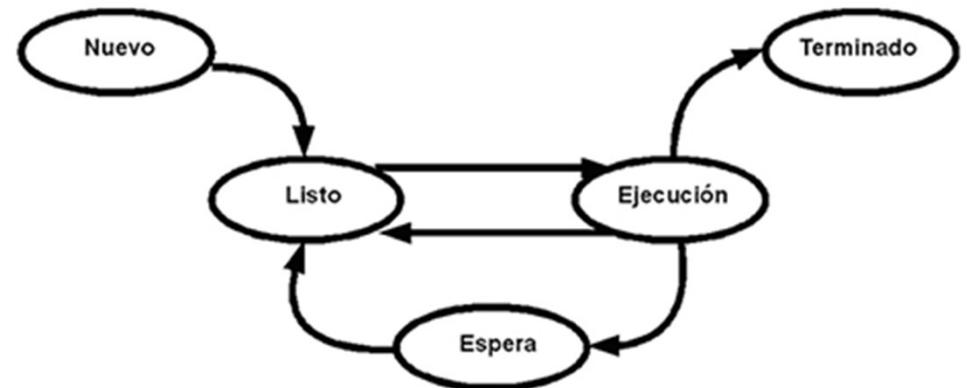
- Un proceso que lanza el sistema operativo sin intervención de ningún usuario está en segundo plano. Son los denominados **demonios** en Linux.
- En los entornos gráficos se considera que están en segundo plano, estrictamente, los procesos que no tienen ventanas asociadas (pronto descubriremos que éstos son muchos).

# ESTADOS DE LOS PROCESOS

Un proceso es un elemento dinámico que va evolucionando. Los estados en los que puede encontrarse son:

**Nonato:** el proceso existe, pero no es conocido por el sistema operativo.

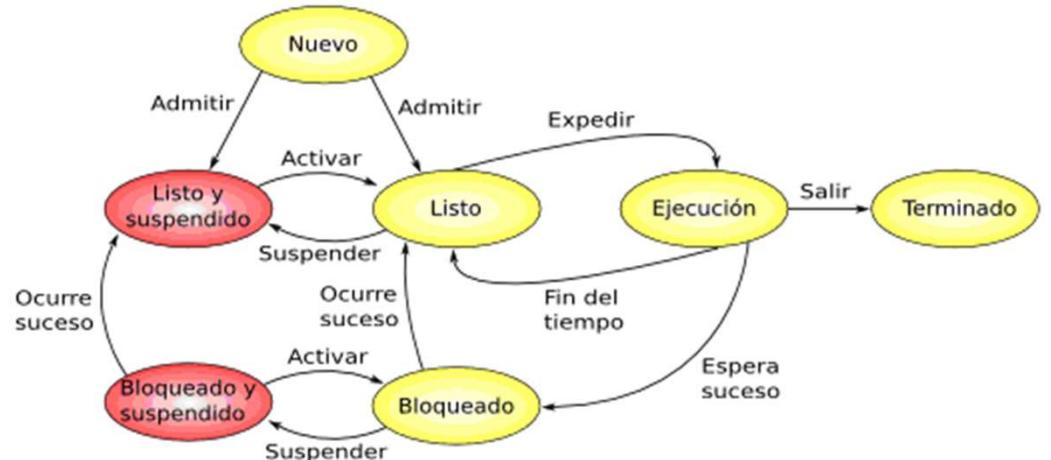
- **Nuevo** o creado.
- Preparado o **listo**.
- Bloqueado, suspendido o dormido (**espera**).
- En ejecución (**running**) o activo.
- Finalizado, muerto o **terminado**.



# ESTADOS DE LOS PROCESOS

Un proceso es un elemento dinámico que va evolucionando. Los estados en los que puede encontrarse son:

- Nonato: el proceso existe, pero no es conocido por el sistema operativo.
- Nuevo o creado.
- Preparado o listo.
- Bloqueado, suspendido o dormido (espera).
- En ejecución (running) o activo.
- Finalizado o muerto.



# ALGORITMOS DE PLANIFICACIÓN

Planificación: cuándo deben asignarse los procesadores a qué procesos

Los procesos tiene prioridades (asignadas por S.O. o el administrador del sistema). →  
Tiempo de ejecución

Objetivos de la planificación:

- Ser justa. Que todos los procesos sean tratados de la misma manera.
- Maximizar la capacidad de ejecución. Dar servicio al número máximo posible de procesos.
- Maximizar el número de usuarios.
- Ser predecible. Debe ejecutarse en la misma cantidad de tiempo y costo con independencia de la carga del sistema.
- Minimizar la sobrecarga. Se trata sobre todo de mejorar el rendimiento total del sistema.
- Equilibrar el uso de recursos.
- Lograr un equilibrio entre respuesta y utilización. Garantizar buenos tiempos de respuesta mediante la disposición de recursos suficientes cuando son necesarios.
- Asegurar prioridades.
- Dar mejor tratamiento a los procesos que muestren un comportamiento deseable.

## CARACTERÍSTICAS DE LOS ALGORITMOS PLANIFICACIÓN

- **Tiempo útil (U) o real de ejecución.** Es el tiempo de uso de la CPU.
- **Eficiencia (EF), eficacia o utilización.** Se expresa como un porcentaje del tiempo medio de utilización, es decir, el porcentaje de tiempo en el que el procesador está ocupado.  $EF=(U/T)*100\%$
- **Rendimiento o Productividad (P).** Es una medida del número de procesos completados por unidad de tiempo. Indica la cantidad de trabajo que se está llevando a cabo. Si N es el número de procesos completados en S segundos.  $P= N/S$

# CARACTERÍSTICAS DE LOS ALGORITMOS PLANIFICACIÓN

- **Tiempo de espera (E).** Es el tiempo que el proceso espera hasta que se le concede el procesador. Puede resultar una medida más adecuada de la eficiencia del sistema, ya que se elimina de la medida el tiempo que tarda en ejecutarse él mismo.
- **Tiempo de retorno o regreso (R).** Es el intervalo de tiempo que transcurre desde que un proceso se crea o presenta hasta que se completa por el sistema o finaliza. Es la suma del tiempo de ejecución real o útil y el tiempo consumido en la espera por los recursos.  
 $R=E+U$
- **Tiempo de respuesta.** Se denomina así al intervalo de tiempo que transcurre desde que se señala un evento hasta que se ejecuta la primera instrucción de dicho evento, es decir, es el tiempo que se está esperando en el estado de preparado o bloqueado para empezar a ejecutarse.
- **Tiempo de uso de CPU total del proceso (T).**

El criterio de selección de un algoritmo se suele basar en la maximización o minimización de una función de los parámetros anteriores. Por ejemplo, maximizar la eficiencia y rendimiento y minimizar el tiempo de espera.

# PROCESOS – MONITORIZACIÓN

- En Linux los procesos son creados siempre por otros que se denominan **padres** de los primeros
- Estructura en forma de árbol → **Linux task-struct**
- Procesos gestionados por el S.O. que pasan desapercibidos para los usuarios – **demonios (daemon)**
- Uso del comando **ps** en Linux

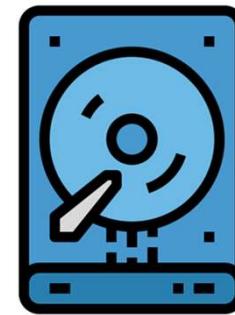
```
jaa@jaa-som:~$ ps aux
USER        PID %CPU %MEM      VSZ   RSS TTY      STAT START  TIME COMMAND
root          1  0.1  0.3 168004 13272 ?      Ss  10:08  0:06 /sbin/init s
root          2  0.0  0.0      0      0 ?      S   10:08  0:00 [kthreadd]
root          3  0.0  0.0      0      0 ?     I<  10:08  0:00 [rcu_gp]
root          4  0.0  0.0      0      0 ?     I<  10:08  0:00 [rcu_par_gp]
root          5  0.0  0.0      0      0 ?     I<  10:08  0:00 [netns]
```

# ALMACENAMIENTO DE LA INFORMACIÓN

Para almacenar información a largo plazo:

- La información debe **permanecer** cuando el proceso termina.
- Se debe poder almacenar una **cantidad grande** de información.
- Debe ser posible que varios procesos tengan **acceso concurrente** a la información.

## Sistema de Archivos



# SISTEMAS DE ARCHIVOS

- El sistema de archivos es la manera de almacenar la información de forma permanente en los soportes externos
- Cada S.O. tiene su **propio** sistema de archivos
- Una sola partición primaria o unidad lógica puede usar **sólo un** sistema de archivos
- Debe presentar la información de forma uniforme **independientemente del dispositivo** usado



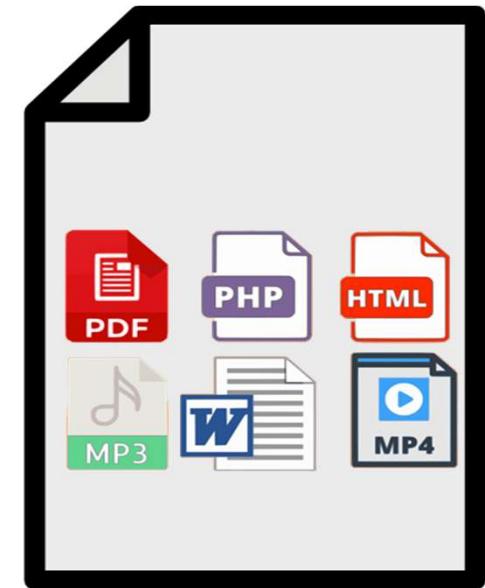
# ARCHIVOS

Un conjunto de información relacionada definida por su creador

- DATOS
- METADATOS

Aspectos a tener en cuenta:

- La longitud máxima del nombre del archivo está limitado
- La longitud máxima de la ruta está limitada
- Se distinguen mayúsculas de minúsculas

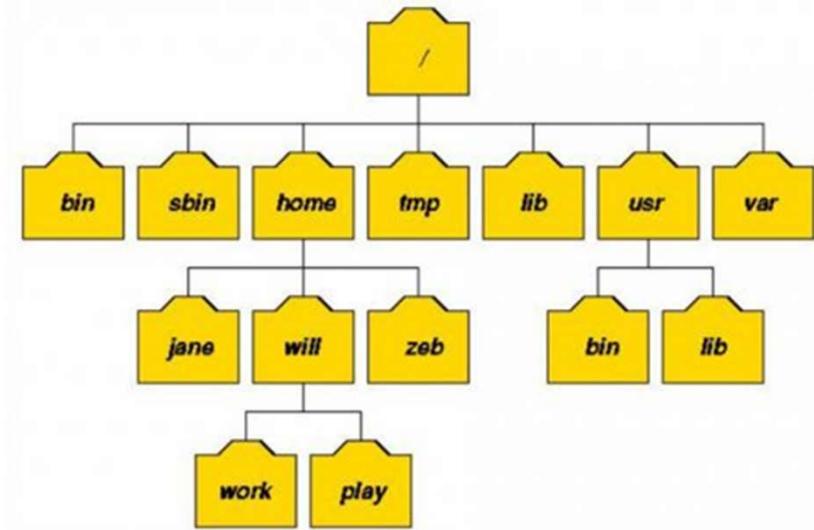


# DIRECTORIO

Es una contenedor que sirve para almacenar archivos u otros directorios

Aspectos a tener en cuenta:

- Los directorios se ordenan de forma jerárquica en forma de árbol invertido
- Un directorio dentro de otro se le denomina **subdirectorio**
- Al directorio que contiene otro directorio se le denomina directorio **padre** del primero
- Al directorio del que cuelgan todos se le denomina **raíz**
- El camino para llegar de una zona del árbol a otra la denominamos **RUTAS**



# ATRIBUTOS

**Información asociada a un archivo o directorio:**

- **Fecha de creación**
- **Fecha del último acceso**
- **Propietario**
- **Tipo de archivo (archivo de sistema)**
- **Permisos del archivo (p.e., sólo lectura)**
- ...

# SISTEMAS DE ARCHIVOS - ATRIBUTOS

En Linux, en la línea de comandos ls -l / ls -al:

```
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Descargas  
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Documentos  
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Escritorio  
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Imágenes  
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Música  
drwxr-xr-x 2 clase clase 4096 may 13 00:45 Plantillas
```

d: atributo de tipo de archivo.

Indica si es un directorio (d), un archivo (-), un enlace (l), etc.

r: Indica que se puede leer.

w: Indica que se puede escribir.

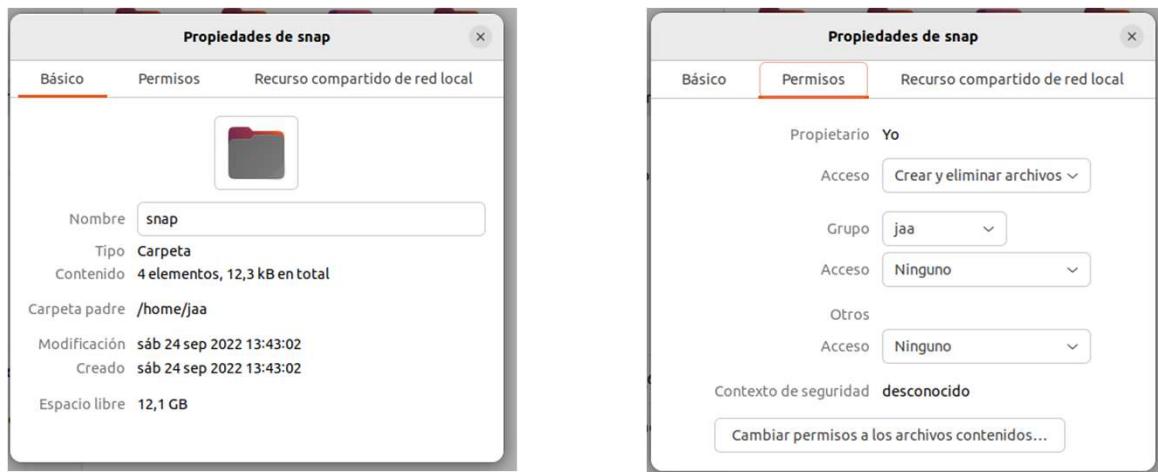
x: Indica que puede ser ejecutado.

Fecha: es el atributo que almacena la fecha de creación o modificación del archivo.

# SISTEMAS DE ARCHIVOS - ATRIBUTOS

En Linux, en el entorno gráfico:

- Explorador de archivos → clic derecho → propiedades



# SELECCIÓN DE UN SISTEMA DE ARCHIVOS

El problema de la asignación del espacio del disco a los archivos es equivalente al de la gestión de la memoria principal.

**Estrategias para almacenar un archivo de n bytes:**

- Asignar n bytes **consecutivos** del espacio del disco.
- Dividir el archivo en bloques que **no adyacentes**.

# SELECCIÓN DE UN SISTEMA DE ARCHIVOS

**En la asignación del espacio del disco los métodos más usados son:**

- Asignación contigua o adyacente.
- Asignación en forma de lista ligada.
- Mediante listas enlazadas o ligadas y un índice.
- Mediante i-nodos.

## I-NODOS

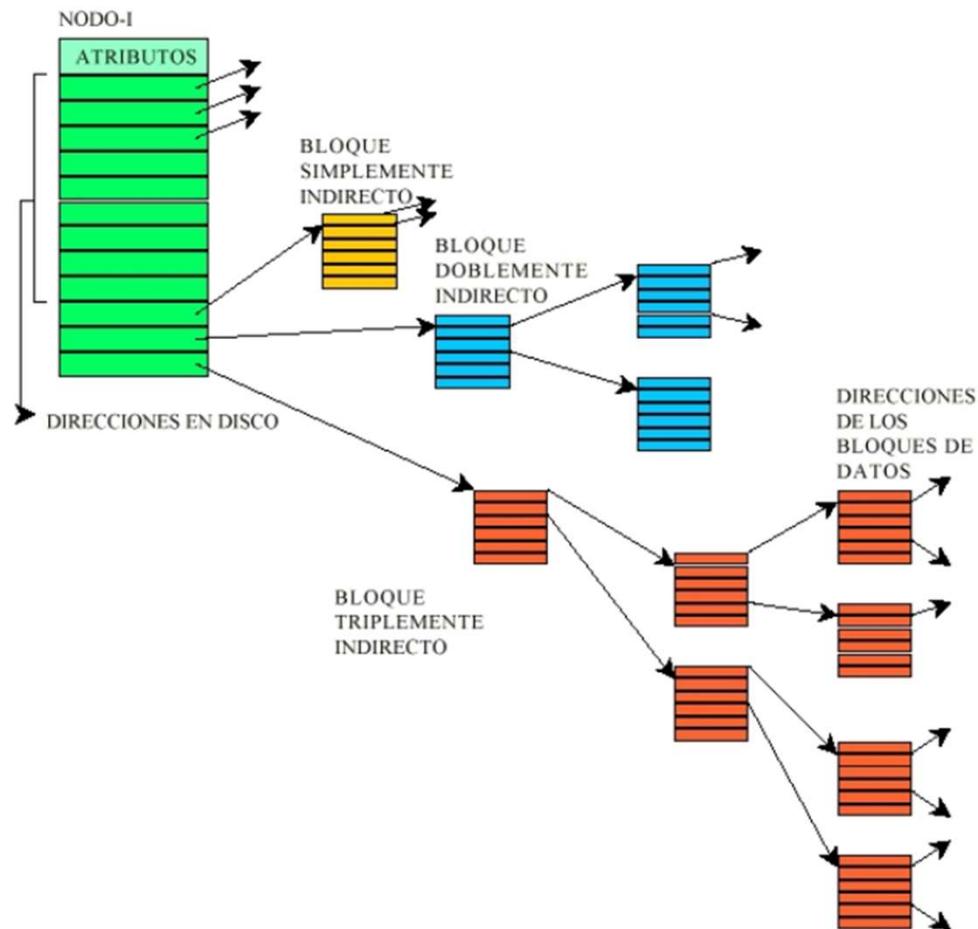
Listas de bloques de diferentes archivos en lugares distintos -> UNIX

Para cada archivo existe una pequeña tabla almacenada en disco que tiene información sobre el archivo y 10 posibles punteros a bloques.

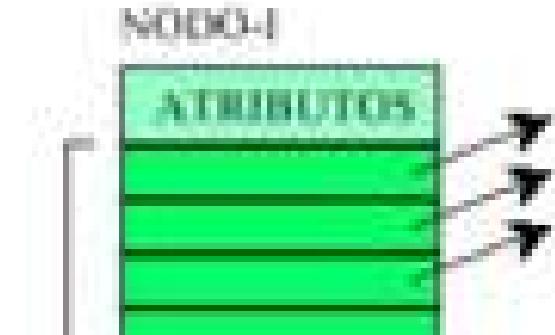
## I-NODOS

- Si archivo < 10 bloques: Suficiente
- Si archivo > 10 bloques: 3 punteros adicionales para redireccionamiento, que permiten hasta un 3er nivel

# I-NODOS



# Atributos del archivo



- Tipo de archivo: regular file, directory, link, ...
- Tamaño: número de bloques asignados al archivo
- Información de protección: propietario, permisos, ...
- Información de tiempo: fecha de creación, último acceso, ...

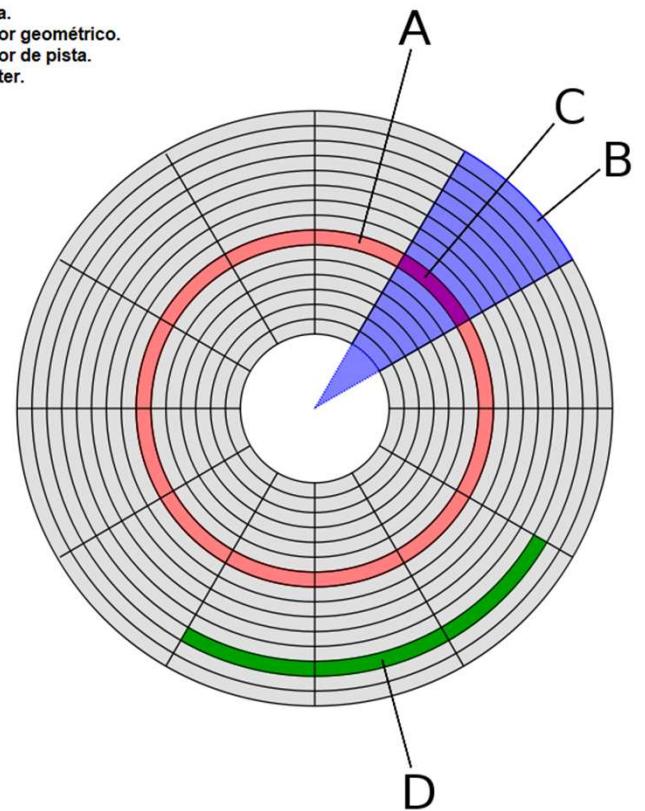
LIMITACIÓN DE I-NODOS

EJERO2

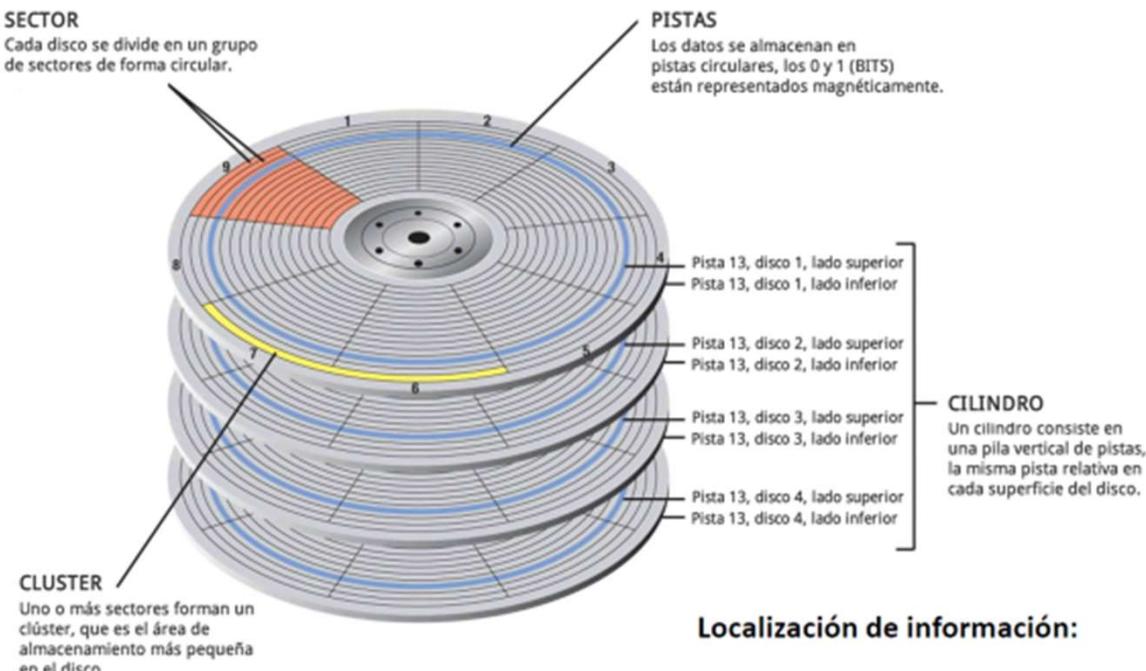
# PARTES DE UN DISCO DURO

- **SECTOR:** mínima unidad de datos en la que está dividido en disco
- **CLÚSTER:** Agrupación de sectores contiguos. Unidad mínima de asignación.
- **BLOQUE:** Mínima unidad de datos direccionable. Un fichero se almacena en el disco cogiendo sectores libres del disco duro.

A = pista.  
B= sector geométrico.  
C= sector de pista.  
D= clúster.



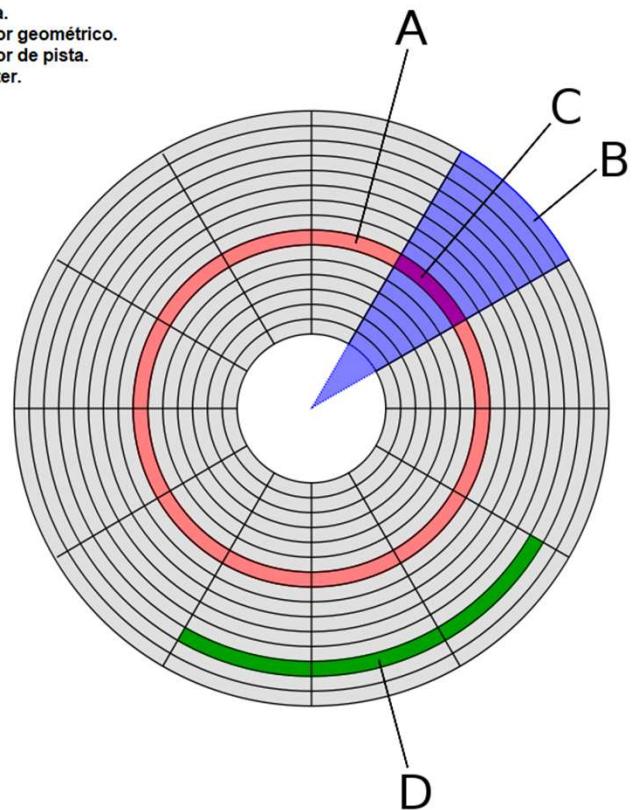
# PARTES DE UN DISCO DURO



# GESTIÓN DEL ESPACIO

- Técnica 1: bloques adyacentes
- Técnica 2: lista ligada
- Técnica 3: lista ligada + índice
- i – nodos:

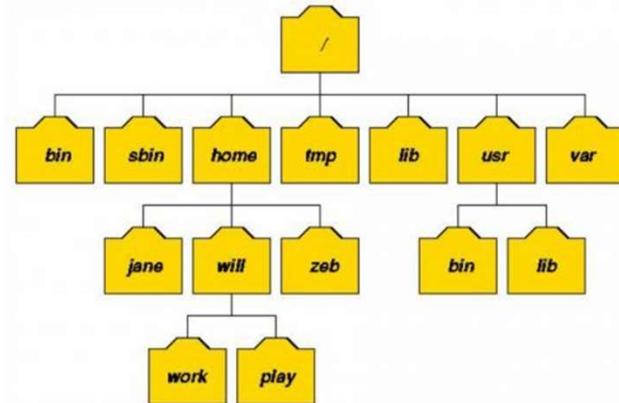
A = pista.  
B= sector geométrico.  
C= sector de pista.  
D= clúster.



# SISTEMAS DE ARCHIVOS

El sistema de ficheros es la **forma** en la que el sistema operativo **almacena y organiza** la información en archivos y directorios dentro de cada unidad lógica de almacenamiento.

Según su complejidad, tienen características como previsión de apagones, posibilidad de recuperar datos, indexación para búsquedas rápidas, reducción de la fragmentación para agilizar la lectura de los datos, etc.



# SISTEMAS DE ARCHIVOS

Existen diferentes sistemas de archivos, los más comunes para Linux y Windows son:

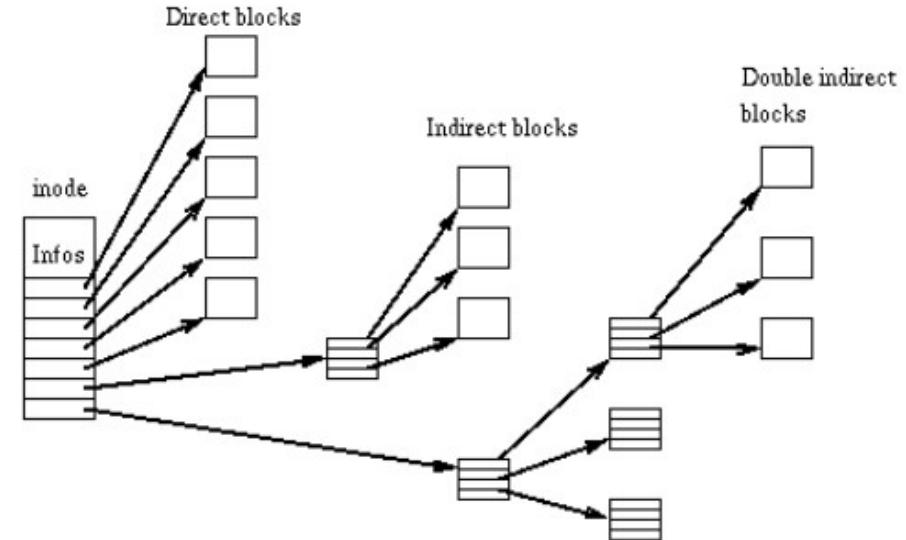
| Linux    | Windows |
|----------|---------|
| ext2     | FAT     |
| ext3     | FAT32   |
| ext4     | NTFS    |
| ReiserFS |         |
| Swap     |         |

# Ext24

**ext2:** alto rendimiento, baja fragmentación, estabilidad y actualización

**ext3:** ext2 + journaling

**ext4:** ext3 + extent

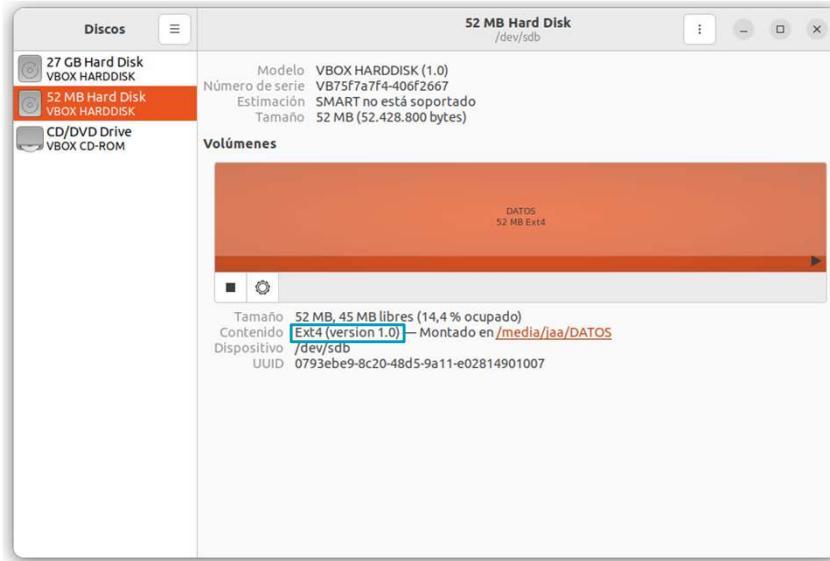


Comprobación del i-nodo que ocupa un fichero: Uso del comando **ls -li**

FILE SYSTEM – ext4

# EJER03

# IDENTIFICAR SISTEMA DE ARCHIVOS EN UBUNTU



```
jaa@jaa-som:~$ df -Th
S.ficheros  Tipo  Tamaño Usados  Disp Uso% Montado en
tmpfs      tmpfs   393M   1,5M  391M  1% /run
/dev/sda3   ext4    24G    12G   12G  51% /
tmpfs      tmpfs   2,0G     0   2,0G  0% /dev/shm
tmpfs      tmpfs   5,0M   4,0K   5,0M  1% /run/lock
/dev/sda2   vfat    512M   5,3M  507M  2% /boot/efi
tmpfs      tmpfs   393M   2,4M  391M  1% /run/user/1000
/dev/sdb    ext4    43M    32K   40M  1% /media/jaa/DATOS
```

# CLASIFICACIÓN DE LOS SO – por los SERVICIOS OFRECIDOS

| Respecto al número de tareas   |   |
|--|---|
| <b>Monotarea</b>   | <b>Multitarea</b>   |
| Solo permite una tarea a la vez por usuario. Estos sistemas solo pueden ejecutar las tareas de una en una.<br>Ejemplo: MSDOS | Permiten al usuario realizar varias tareas al mismo tiempo. Se distinguen por su capacidad para soportar la ejecución concurrente de dos o más procesos.<br>Ej: Microsoft Windows y Apple MacOS |
| Respecto al número de usuarios   |   |
| <b>Monousuario</b>   | <b>Multiusuario</b>   |
| Solo se puede atender a un único usuario a la vez.<br>Ej: MSDOS, CM/P, Windows 3.1   | Soportan el trabajo de varios usuarios a la vez, y pueden compartir recursos. Se emplean especialmente en redes.<br>Ej: UNIX, Linux, Windows 7, Windows XP, etc.                                |
| Respecto al número de servicios  |   |
| <b>Uniproceso</b>  | <b>Multiproceso</b>   |
| Capaz de manejar un solo procesador.<br>Ej: MSDOS y MACOS  | El ordenador cuenta con varios procesadores, y el sistema puede usarlos todos para distribuir su carga de trabajo.<br>Ej: Solaris, SCO Unix   |

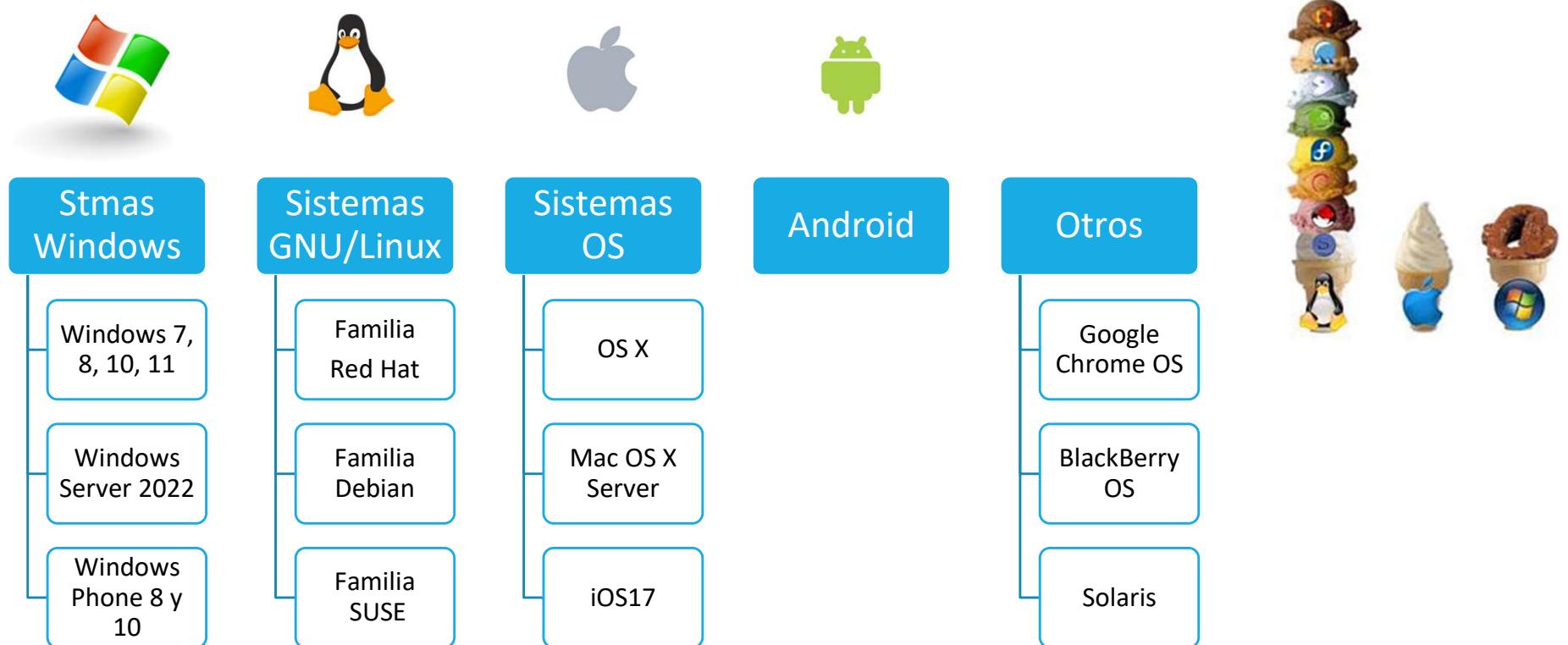
# CLASIFICACIÓN DE LOS SO – SEGÚN MODO DE OFRECER LOS SERVICIOS

| Respecto al <b>ACCESO</b> del usuario a sus servicios   |  |
|---|--|
| De red  | Distribuido  |
| <p>Conectan dos o más ordenadores a través de algún medio de comunicación con el objetivo de <u>compartir los diferentes recursos y la información del sistema</u>. El usuario debe saber la sintaxis de los comandos y llamadas, además de la ubicación de los recursos a los que se desea acceder.</p> <p>Ej: Novel Netware, Windows Server, Linux o UNIX</p> | <p>En los sistemas distribuidos existe un conjunto de ordenadores conectados entre si de forma que los usuarios acceden a todos los recursos de todos los ordenadores como si fuese un servicio integrado. No necesita saber la ubicación de los recursos, los conoce por nombre y <u>los usa como si todos ellos fuesen locales</u>. Están diseñados para que muchos usuarios trabajen de forma conjunta.</p> <p>Ej: Solaris-MC, Mach o Chorus.</p> |

# Clasificación de SO: por su ESTRUCTURA INTERNA

- **MONOLÍTICO:** Un único gran programa (kernel) trabaja en modo privilegiado
- **CAPAS:** Organización jerárquica con distintos privilegios
- **MICROKERNEL:** subdivisión en módulos en los que solo uno, el microkernel, que opera en modo privilegiado
- **CLIENTE/SERVIDOR:** procesos que ofrecen servicios y procesos que los usan
- **MÁQUINA VIRTUAL (ud2)**
  - Hipervisor tipo 1 o nativo: ejecución sobre el HW
  - Hipervisor tipo 2 o alojado: ejecución como una aplicación sobre el SO

# SISTEMAS OPERATIVOS ACTUALES



"Linux vs Mac vs Windows" by KobraSoft is licensed under CC BY 2.0

# DISTRIBUCIONES

- **KERNEL:** compartido
- **FAMILIAS:** Debian, Red Hat, SUSE... Ubuntu
- **APLICACIONES:** cada familia tiene aplicaciones específicas para cada **uso**
- **GESTIÓN DE PAQUETES:** rpm, deb, tgz...
- **LiveCD, LTS, LITE**



<https://distrowatch.com/>

# SISTEMA OPERATIVO





## UD1. INTRODUCCIÓN A LOS SISTEMAS OPERATIVOS.

¿Preguntas?