UD 4. Sistemas Operativos

# Clasificación del software según su tipo

**Software**: conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.

Se puede clasificar en jerarquía dependiendo de su cercanía al hardware o al usuario.

## Software de sistema

**Software de sistema**: conjunto de software que se encarga de gestionar los recursos hardware del sistema informático, sirviendo de base a la ejecución de los programas de aplicación. Actúa de intermediario entre el hardware del equipo y los usuarios del mismo. Intermediación:

* Usuarios: interfaz de usuario, gráfica o Terminal.
* Aplicaciones: “API”, conjunto de funciones y procedimientos.

El principal es el S.O., aunque se incluyen controladores, herramientas de diagnósticos y otras utilidades.

## Software de aplicación

Permiten a los usuarios realizar tareas concretas, que pueden ser:

* Generales: procesadores de texto, navegadores…
* Específicas: contabilidad, videojuegos…

Junto al S.O. se suelen instalar algunas aplicaciones que los proveedores incorporan como complemento para facilitar ciertas tareas.

## Lenguajes de programación

**Lenguaje de programación**: conjunto de símbolos y reglas sintácticas y semánticas diseñadas para crear programas.

Actualmente son de **alto nivel**, pero se necesita **compilar** para ser comprendidos por el ordenador.

Los programadores utilizar **Entorno de Desarrollo Integrados** (IDE) para programar. Incluyen:

* Editor para escribir código fuente.
* Compilador para convertirlo a código objeto.
* Otras herramientas, como depuradores.

# Clasificación del software según su licencia

## Tipos de licencias

**Licencia**: contrato entre el proveedor de un programa informático y el usuario (o empresa) que lo utiliza, para fijar las condiciones a las que se obligan ambas partes durante el tiempo en el que el programa esté en uso. Entre las regulaciones de una licencia de software, están:

* Vinculación de un programa a un determinado hardware.
* Numero de copias que puede utilizar el usuario.
* Derechos del usuario sobre el programa.
* Periodo durante el que se tendrán los derechos.
* Responsabilidad del proveedor sobre los fallos.
* Posibilidad de ceder el programa a otras personas.

Dependiendo de los derechos del proveedor sobre su producto, están las siguientes licencias:

* **Licencias de código abierto**: el código fuente está a disposición de los usuarios. Unido al concepto de *software libre*. Subtipos:
  + **Permisivas**: los programas se pueden modificar o crear otros nuevos **sin que el resultado tenga que mantener la licencia original**.
  + **No permisivas**: los programas se pueden modificar o crear otros nuevos, pero el resultado **debe tener los mismos términos de la licencia original**.
* **Licencias de código cerrado**: el código fuente no se distribuye con el software. Se conoce como *software privado*. Se limitan las posibilidades para utilizar, copiar, modificar, redistribuir o ceder el software. Licencias **CLUF** (Contrato de Licencia para el Usuario Final) o **EULA** (en inglés).

Variantes de tipos de licencias:

* **Freeware**: se distribuye gratuitamente por un tiempo indefinido.
* **Shareware**: permite probar el software, pero puede limitar el tiempo o algunas funcionalidades.
* **Dominio público**: sin licencia. Se puede utilizar, modificar, redistribuir o licenciar sin limitaciones.

## Libertades del software libre

Definición asociada a la *Free Software Fundation* (1985). Para que un programa sea *software libre*, tiene que garantizar las siguientes libertades:

* **Libertad 0 (uso)**: usar el programa con cualquier propósito.
* **Libertad 1 (estudio)**: estudiar y modificar el programa adaptándolo a las propias necesidades.
* **Libertad 2 (distribución)**: distribuir copias del programa para ayudar a otros usuarios.
* **Libertad 3 (mejora)**: mejorar el programa y hacer públicas las mejoras.

## Comercialización de productos con licencias propietarias

Pueden adquirirse a través de diferentes vías:

* **Retail**: forma habitual de comprar software en un establecimiento. Dirigida a usuarios que necesitan pocas licencias y tiene 2 variantes:
  + **Producto completo**: no necesita versión previa para instalar.
  + **Actualización**: parte de una versión previa (con licencia).
* **OEM**: preinstalado en un ordenador. Garantiza correcta instalación y configuración. Vinculada al hardware.
* **Licencias por volumen**: dirigidas a empresas o entidades. Derechos como transferencia a determinados usuarios. Opción de producto completo o actualización. Suelen disponer de derechos de *downgrade*.

## Ventajas y desventajas del software libre

**Ventajas**:

* Mas económico o sin coste.
* El soporte puede ser local, así no se depende de una compañía extranjera.
* Se pueden instalar tantas veces y en tantos equipos como sea necesario. Se puede compartir libremente.
* Los formatos abiertos facilitan la interoperabilidad entre sistemas.
* Acceder al código fuente permite la creación de nuevos productos sin necesidad de empezar de cero.
* Ayuda a la reducción de brechas tecnológicas entre países.
* Los fallos de seguridad son resueltos por la comunidad en plazos muy cortos.
* El modelo de negocio se suele basar en el servicio y no en las licencias.
* No existe la “obsolescencia programada”.
* Mas seguridad de que no haya código malintencionado o no autorizado.
* Los repositorios permiten acceso a miles de aplicaciones y otras herramientas, revisadas y garantizadas, de forma gratuita.

**Desventajas:**

* El acabado estético suele ser inferior al software propietario.
* Hay muchas aplicaciones propietarias que son un estándar.
* Los videojuegos suelen estar solo disponibles para Windows.
* Suele haber mayor desconocimiento del funcionamiento.
* Existe hardware que solo funciona en Windows por compatibilidad de drivers.
* Existe un mercado laboral inferior, aunque más especializado.

# Sistemas Operativos

## Tipología de los SS.OO.

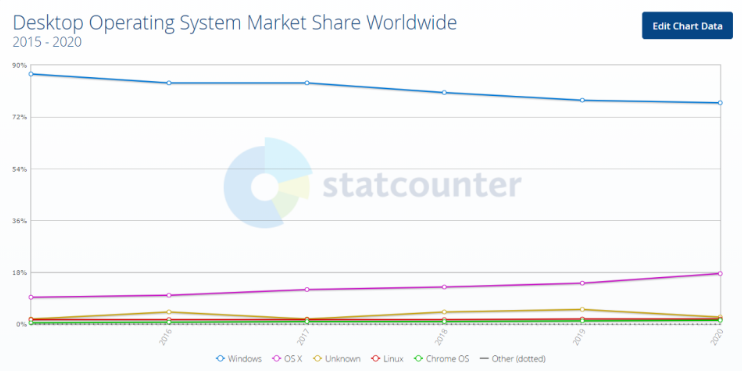
Los Sistemas Operativos tienen dos misiones:

* Administrar hardware.
* Actuar como interfaz “sencilla” de usuario (texto o gráfica).

Hay varias maneras de clasificarlos:

* Monousuario o multiusuario: usuarios que puedan usar a la vez el S.O.
* Monopuesto o multipuesto: personas que interactúan a la vez mediante Terminales.
* Monotarea o multitarea: tareas de una en una, o en paralelo o forma concurrente.
* Monoprocesador o multiprocesador: una o varias CPUs.
* Monoprogramado o multiprogramado: uno o varios programas en memoria al mismo tiempo.

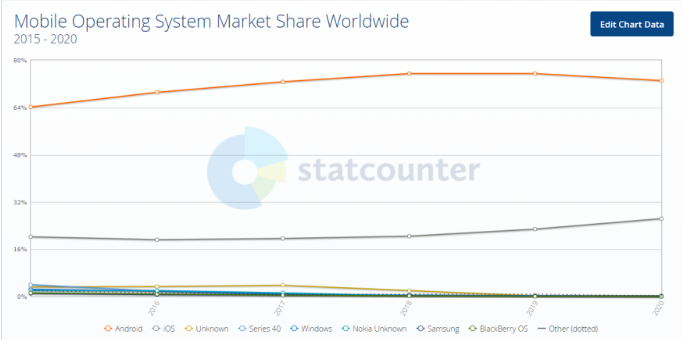
## Panorámica actual de los SS.OO.



**Ordenadores servidores:**

* Microsoft Windows Server.
* GNU/Linux.
* UNIX.
* Solaris/OpenSolaris.
* Apple OS X Server.

**Ordenadores móviles:**

****

**Otros sistemas**:

* Chrome OS: basado en GNU/LINUX, desarrollado por Google y orientado a Internet.
* webOS: basado en GNU/LINUX, desarrollado por LG para sus televisores.
* Tizen: basado en GNU/LINUX, para ser instalado en teléfonos.
* Sistemas empotrados: Android, iOS y Windows tienen versiones para implantar en dispositivos como smartwatches, coches… intentado orientarse al “Internet de las cosas”.

## SS.OO. libres y propietarios (escritorio)

Hay 3 principales familias de SS.OO.:

* Microsoft Windows: principal de los sistemas con licencia privativa.
* Linux Ubuntu: el de mayor repercusión con licencia de código abierto.
* Mac OS: innovador, orientado a un hardware concreto, con licencia privativa.

### Microsoft Windows

Dispone de una gama de SS.OO. que cubren teléfonos, sistemas empotrados, PCs y servidores.

El primer Windows (1985) era una GUI de MS-DOS intentando parecerse a Mac OS (1984).

A partir de Windows XP, el núcleo de MS-DOS se cambió por uno NT de Windows 2000 (gama profesional).

El siguiente gran cambio vino con Windows 8 con la remodelación de su GUI. Otro gran cambio fue el soporte de procesadores ARM.

### GNU/LINUX

Historia comenzada en **1960** con **Multics**, un proyecto de S.O., aunque se paró. Dos programadores continuaron el proyecto en **1969** rebautizándolo como **Unics**. Finalmente, en **1972**, **UNIX** fue escrito en **C**.

En un principio se puso a disposición de las universidades, donde se desarrollaron variaciones.

En **1983**, **Richard Stallman** anunció **GNU**, que pretendía crear un S.O. libre, aunque se estancó.

En **1991**, **Linus Torvalds** termina el S.O., y lo publica en **1992** con licencia GPL de GNU.

#### “Distros” de GNU/Linux

Al ser de código abiertos, programadores de todo el mundo colaboran en el S.O. y sus aplicaciones.

Han salido varias distribuciones: **Debian**, **Red Hat**, y **Slackware** son las “matrices”.

Una de las más representativas es **Ubuntu**, basada en *Debian* y patrocinada por **Canonical**. Se ofrece en diferentes versionas para ámbitos domésticos y profesionales. Es actualizado cada 6 meses con nombres de animales y un adjetivo (por nombre alfabético). Cada 2 años aparece una versión **Long Time Services** (LTS), que cuenta con soporte de Canonical durante 5 años (el resto durante 9 meses).

### mac OS

anteriormente llamado OS X, es un S.O. basado en UNIX, desarrollado, comercializado y vendido por **Apple Inc**.

A diferencia de los anteriores, mac OS solo se puede ejecutar en un **hardware concreto**. Esto asegura la venta de la máquina y garantiza su eficiencia.

Esta basado en el **núcleo Darwin** e incluye la GUI *Aqua y el Finder*.

Nombra a sus versiones con **lugares de California**.