

# Clase 2

## Software de computación

La definición formal es que es la suma total de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación. El Software es también conocido como el conjunto de programas que puede ejecutar el hardware para la realización de las tareas de computación a las que se destina, esta visión es la indicada cuando hablamos del software, porque debemos vincular este concepto directamente con la importancia que tiene para un sistema de computación.

## Primera aplicación del término

El término software fue usado por primera vez en este sentido por el estadista John W. Tukey 21 en 1957. En la ingeniería de software y las ciencias de la computación, el software es toda la información procesada por los sistemas informáticos: programas y datos. Tukey acuñó muchos términos estadísticos que ahora son de uso común, pero las dos más famosas palabras inventadas por él están relacionadas con la informática. Mientras trabajaba con John Von Neumann en los primeros diseños de computadoras, Tukey introdujo la palabra "bit" como contracción de "Dígito binario" (por sus siglas en inglés Binary Digit).

# Clasificación de Software

## Por distribución

### Software libre

Por definición, el software libre es aquel que puede ser distribuido, modificado, copiado y usado; por lo tanto, debe venir acompañado del código fuente para hacer efectivas las libertades que lo caracterizan.

Dentro de software libre hay, a su vez, matices que es necesario tener en cuenta. Por ejemplo, el software de dominio público significa que no está protegido por el copyright, por lo tanto, podrían generarse versiones no libres del mismo, en cambio el software libre protegido con copyleft impide a los redistribuidores incluir algún tipo de restricción a las libertades propias del software así concebido, es decir, garantiza que las modificaciones seguirán siendo software libre, también es conveniente no confundir el software libre con el software gratuito, este no cuesta nada, hecho que no lo convierte en software libre, porque no es una cuestión de precio, sino de libertad.

Para Richard Stallman el software libre es una cuestión de libertad, no de precio. Para comprender este concepto, debemos pensar en la acepción de libre como en “libertad de expresión”. En términos del citado autor el software libre se refiere especialmente a cuatro clases de libertad para los usuarios de software:

**o Libertad 0:** la libertad para ejecutar el programa sea cual sea nuestro propósito.

**o Libertad 1:** la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades “el acceso al código fuente es condición indispensable para esto”.

**o Libertad 2:** la libertad para redistribuir copias y ayudar así a tu vecino.

**o Libertad 3:** la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad y el acceso al código fuente es condición indispensable para esto.

En 1983 Stallman fundó el proyecto GNU, con el fin de crear sistemas operativos parecidos a UNIX.

Dos años más tarde creó la "Fundación del Software Libre" y escribió la GPL (General Public License) para posibilitar el software libre en el sistema de copyright.

## **Ventajas del Software Libre**

- Existen aplicaciones para todas las plataformas.
- Libertad de copia.
- Libertad de modificación y mejora.
- Libertad de redistribución.
- Facilidad a la hora de traducir una aplicación en varios idiomas.
- Mayor seguridad y fiabilidad.

## **Desventajas del Software Libre**

- Algunas aplicaciones pueden llegar a ser algo complicadas de instalar.
- Menor compatibilidad con el hardware.

## **Software propietario**

Por definición, se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o que su código fuente no está disponible o el acceso a este se encuentra restringido. En el software no libre una persona física o jurídica posee los derechos de autor sobre un software negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito.

Los referentes más conocidos del software propietario son Bill Gates que creó junto a Paul Allen la empresa de Software Microsoft en 1975. Y Steve Jobs que fundó Apple junto a Steve Wozniak en 1976.

Hablando de software propietario, es importante saber también los siguientes términos:

**Licencia de Software:** Contrato entre el desarrollador de un software sometido a propiedad intelectual y a derechos de autor y el usuario, en el cual se definen con precisión los derechos y deberes de ambas partes. Es el desarrollador, o aquél a quien éste haya cedido los derechos de explotación, quien elige la licencia según la cual lo distribuye.

**Patente:** Conjunto de derechos exclusivos garantizados por un gobierno o autoridad al inventor de un nuevo producto (material o inmaterial) susceptible de ser explotado industrialmente para el bien del solicitante por un periodo de tiempo limitado.

**Derecho de autor o copyright:** Forma de protección proporcionada por las leyes vigentes en la mayoría de los países para los autores de obras originales incluyendo obras literarias, dramáticas, musicales, artísticas e intelectuales, tanto publicadas como pendientes de publicar.

**Copyleft:** Es el término que se utiliza en el ámbito informático (se aplica de manera análoga a la creación literaria y artística) para designar el tipo de protección jurídica que confieren determinadas licencias que garantizan el derecho de cualquier usuario a utilizar, modificar y redistribuir un programa o sus derivados, siempre que se mantengan estas mismas condiciones de utilización y difusión.

## **Ventajas del Software Propietario**

- Facilidad de adquisición
- Las empresas que desarrollan este tipo de software son por lo general grandes y pueden dedicar muchos recursos, sobre todo económicos, en el desarrollo e investigación.

## **Desventajas del Software Propietario**

- No existen aplicaciones para todas las plataformas.
- Imposibilidad de copia y redistribución.
- Imposibilidad de modificación.
- El costo de las aplicaciones es mayor.
- El usuario que adquiere software propietario depende al 100% de la empresa propietaria.

## **Por funciones que cumplen**

### **Software del sistema**

Junto con el sistema de computación se vende o entrega el denominado Software del Sistema (o parte del mismo) sin el cual su manejo sería bastante complicado para la realización de las tareas requeridas, y su programación estaría a cargo de especialista en el hardware, también por seguridad no conviene que usuarios no especializados accedan a este tipo de información. Este software se encuentra almacenado en la memoria ROM y se lo conoce con el nombre de BIOS (Sistema Básico de E/S), la manera de acceder a este software es por medio del programa Setup, en donde aparece un menú con varias opciones que nos permiten, por ejemplo, reconocer un nuevo dispositivo de almacenamiento. La función principal del software del sistema es reconocer

a los componentes básicos del hardware, a partir de su reconocimiento, los configura, y le brinda a los sistemas operativos un hardware ya reconocido.

## **Software base**

El Software de Base está compuesto de un grupo de programas que son la base para el uso de un sistema de computación, una parte de estos programas, una vez que encendimos el equipo y se ejecutaron los pasos para efectuar la carga del mismo, residen permanentemente en la memoria RAM, mientras el equipo está encendido, el software de base siempre es el mismo. El otro grupo de programas está compuesto de distintas herramientas informáticas que le permitirán a los usuarios realizar el procesamiento de los datos con el programa adecuado para cada tipo de información y colaborar con el sistema operativo en la administración de un sistema de computación. Hay paquetes de programas integrados que sirven en forma completa para la administración óptima de una oficina utilizando un sistema de cómputo y forman parte de lo que se conoce como suite ofimática. Estas suites suelen incluir procesador de texto, hoja de cálculo, gestión de base de datos, cliente de correo electrónico, agenda y administrador de presentaciones o diapositivas (Por ejemplo: Microsoft Office). Estos programas forman parte del software base.

## **Software de aplicación**

El Software de Aplicación está compuesto por un conjunto de programas creados para atender los trabajos específicos del usuario.

Los programas que conforman a este tipo de software reúnen características que se relacionan únicamente con la aplicación para la cual fueron concebidos, como por ejemplo: control de stock, contabilidad general, sueldos y jornales, cuentas corrientes, deudores, proveedores, etc.

Para el desarrollo de los programas que forma parte de este tipo de software se deben tener en cuenta las distintas metodologías de procesamiento de datos que va ser utilizada.

## **Por administración**

### **Administración de tareas**

**Monotarea:** pueden ejecutar sólo un proceso por vez y son usados en hardware de uso específico. Por ejemplo, MS-DOS.

**Multitarea:** permite ejecutar muchas tareas de manera simultánea.

### **Administración de usuarios**

**Monousuario:** son versiones de sistemas operativos en dónde sólo hay oportunidad de crear un sólo usuario (como por ejemplo, las versiones viejas de Windows anteriores al 9x).

**Multiusuario:** permiten tener a más de un usuario ejecutando diferentes tareas de manera simultánea utilizando los mismos recursos físicos de la computadora.

## Administración de recursos

**Centralizados:** todas las ejecuciones de instrucciones ocurren en una sola computadora.

**Descentralizados:** hay un equipo base que mantiene un programa funcionando y puede ser, a la vez, manejado por otros dispositivos.

# Arquitectura de los sistemas operativos

## Kernel

El Kernel es el corazón del sistema operativo. Esencialmente comunica el hardware con el software. Organiza los procesos y los datos. Permite que haya movimiento de datos en la computadora. También controla todos los accesos al procesador y la memoria, pero también es responsable de los drivers.

Cuando un dispositivo de entrada ingresa datos a la computadora, el Kernel recibe una “llamada al sistema” y va procesando toda la información que recibe del usuario y la convierte en lenguaje máquina para que lo entienda y reciba a su vez el procesador.

La estructura del Kernel tiene como capa más baja la interfaz con el hardware, donde está el procesador, memoria y dispositivos.

Por encima tenemos una capa en donde, por un lado, está el controlador de hardware, el gestor de procesos que se encarga de la gestión del tiempo y la que permite que se puedan realizar varias tareas a la vez, y el gestor de memoria que se dedica a distribuir datos entre la memoria RAM y la memoria virtual.

En la capa más alta está el sistema de archivos.

El Kernel acompaña y sirve como guía durante todo el recorrido de la orden que le das desde el hardware hasta el software de la aplicación.

Para redondear, las funciones del Kernel son:

- Gestión de memoria: analiza cuánta memoria es utilizada para almacenar distintos elementos y también en qué lugar hay que guardarlos.
- Gestión de procesos: establece cuáles procesos pueden usar la unidad de procesamiento, cuándo y por cuánto tiempo.
- Controladores de dispositivos: es un puente entre el hardware y los

procesos que se ejecutan.

- Seguridad y llamadas al sistema: recibe las diferentes solicitudes de servicio por parte de los procesos.
- Protección: otorga y niega permisos a software de dudosa procedencia.
- Gestión de dispositivos externos: todos los dispositivos quedan sujetos a la gestión del Kernel.
- Gestión de optimización: cómo se ejecutan diferentes procesos a la vez, establece prioridades y elimina las que le parecen irrelevantes.

## **Tipos de Kernel**

### **Microkernel:**

Es una arquitectura de sistema operativo donde el núcleo proporciona sólo las funciones básicas de gestión de hardware y comunicación entre procesos. Las funciones más avanzadas, como sistemas de archivos y gestión de memoria, se ejecutan como procesos independientes fuera del núcleo. Esto hace que el sistema sea más modular y flexible, pero puede tener un impacto en el rendimiento. Ejemplo: MINIX.

### **Monolítico:**

En contraste con el microkernel, un núcleo monolítico incorpora todas las funcionalidades del sistema operativo en un solo ejecutable. Esto incluye la gestión de memoria, el sistema de archivos, los controladores de dispositivo, etc. Es más rápido en términos de rendimiento, pero puede ser menos flexible y más difícil de mantener y depurar. Ejemplo: Linux. Aunque Linux tiene la capacidad de cargar módulos de forma dinámica para extender su funcionalidad, sigue siendo considerado monolítico debido a que la mayoría de sus funcionalidades principales están integradas en el mismo núcleo. Ejemplo: Linux y Algunas Mac

### **Híbrido:**

Un sistema operativo híbrido combina características de ambos enfoques. Por ejemplo, puede tener un núcleo monolítico que proporciona las funciones básicas del sistema operativo, pero permite que partes específicas se ejecuten como módulos o procesos independientes, similar a un microkernel. Esto busca combinar la eficiencia del monolito con la flexibilidad del microkernel. Ejemplo: Windows. y Algunas Mac (XNU)

## Sistema de archivos

Es el que controla cómo se almacenan y recuperan los datos. Asigna espacio a los archivos, administra el espacio libre y el acceso a los datos guardados. Pone en estructura la información guardada en el disco duro para ser después representada textual o gráficamente en el gestor de archivos.

En Linux, se usa un sistema de archivos jerárquico que empieza con el directorio raíz (/).

En cambio, en Windows, la estructura en unidades. (empieza por la unidad C: y llega hasta la Z:).

## Interfaz gráfica

Es una especie de traductor en la comunicación entre el humano y la máquina. Permite que cualquier usuario, independientemente del conocimiento que tenga de una computadora, pueda entender de manera rápida los programas. Los íconos son fáciles de entender y los textos explicativos facilitan el uso.

Las ventajas de la interfaz gráfica son que son simples, fáciles de usar, con diseños atractivos. Facilitan la búsqueda de documentos y archivos.

Las desventajas es que sólo se pueden ejecutar instrucciones programadas. Requieren además mucha memoria y son más lentos que las interfaces basadas en líneas de comando.