



# Tema 4: Software de un Ordenador

## Informática

**Departamento de Informática**  
**Israel González Carrasco**  
**(israel.gonzalez@uc3m.es)**  
**María Belén Ruiz Mezcua**  
**(mbelen.ruiz@uc3m.es)**

# Contenido

---

1. Software.
2. Tipos de software.
3. Sistemas operativos.
4. Lenguajes de programación.
5. Ingeniería del software.
6. Enfermedades del software.
7. Seguridad informática.

# Introducción al Software

- Lenguaje
  - Forma de escribir las instrucciones en el ordenador.
  - Al principio de la aparición de la informática se hacía directamente con lenguajes de bajo nivel o lenguaje máquina (0,1).
- Código máquina
  - Sistema de códigos directamente comprensibles por la UCP.
  - Posteriormente se pierde la obligatoriedad de escribir en lenguaje binario, aparecen los ensambladores que utilizan un formalismo cercano al simbolismo matemático.
  - Al evolucionar llegan a los lenguajes de alto nivel próximos al lenguaje natural del hombre como Cobol, Fortran, Basic etc.
  - Estos lenguajes de comunicación con la máquina se llaman lenguajes de programación.

# Introducción al Software (II)

- Instrucciones
  - Son órdenes para realizar una acción.
- Programa
  - Conjunto de instrucciones escritas en secuencia que ejecutadas en un ordenador, permiten la resolución de una tarea concreta. El programa lo escribe un técnico informático o programador en un lenguaje creado para este propósito.
- Software
  - Conjunto de programas que propician el correcto funcionamiento del ordenador y de los periféricos, por una lado, y la resolución de problemas generados por el usuario, por otro.

# Introducción al Software (III)

- Lenguaje de Programación
  - Conjunto de reglas sintácticas y semánticas para indicar a un computador las instrucciones que debe ejecutar, en forma de programas.
- Aplicación Informática
  - También se denominan programas de aplicación. Su misión consiste en resolver tareas finales muy variadas que van desde la gestión empresarial, el cálculo científico o técnico, o paquetes de ofimática, hasta programas de educación o de juegos.
  - El proceso de elaboración de una aplicación se hace en varias etapas.

# Tipos de Software

- Denominación general.
  - Shareware.
    - ✓ Tras un período de prueba sin coste, pueden adquirirse posteriormente a bajo precio para poder seguir utilizándolos.
  - Freeware.
    - ✓ Software de uso libre y totalmente gratuito, que puede modificarse y comercializarse.
  - Firmware.
    - ✓ Software implantado en el hardware (operaciones de arranque del ordenador, grabados sobre circuitos electrónicos).

# Tipos de Software (II)

- Software Básico.
  - Conjunto de programas sin los cuales el ordenador no podría funcionar. Algunos programas se encuentran cargados, pero hay otros que se deben incorporar para el buen funcionamiento del equipo.
    - ✓ Sistema Operativo.
    - ✓ Programas de Utilidad: Agregar/quitar archivos.
    - ✓ Programas de comunicaciones: Programas para red.
    - ✓ Programas antivirus. Para eliminar programas malignos (virus, gusanos, troyanos, etc.).

# Tipos de Software (III)

- Software Específico.
  - Programas de Usuario que resuelve problemas concretos de usuarios y que han tenido en cuenta los distintos S.O, así como los requisitos de hardware en lo que se van a ejecutar.
    - ✓ Aplicaciones a medida.
    - ✓ Paquetes integrados de gestión empresarial .
    - ✓ Paquetes de ofimática.
    - ✓ Software educativo.
    - ✓ Programas de multimedia.
    - ✓ Juegos.



# Tipos de Software (IV)

- Software Específico.
  - Aplicaciones a medida: Empresa, Outsourcing.
    - ✓ Programas creados por una empresa dedicada al desarrollo de software.
    - ✓ Aplicaciones relacionadas con áreas básicas o específicas de la empresa.
  - Paquetes integrados de gestión empresarial .
    - ✓ Conjunto de programas enfocados a la gestión de las áreas básicas de PYMES (Pequeñas y Medianas Empresas) y que por ser comunes a la mayoría se suelen encontrar en el mercado.

# Tipos de Software (V)

- Software Específico.
  - Paquetes de ofimática.
    - ✓ Conjuntos de programas que incluye un procesador de textos, una hoja de cálculo, un programa de presentación y una base de datos.
    - ✓ Microsoft Office, OpenOffice, etc.
  - Software educativo.
    - ✓ Programas destinados a la adquisición de conocimientos.
  - Programas de multimedia.
    - ✓ Software que admite el tratamiento digital de todo tipo de información (audio, video, imágenes, texto, etc.).
    - ✓ Enciclopedias.
  - Juegos.
    - ✓ Programas multimedia para diferentes plataformas con fines lúdicos o educativos.

# Sistema Operativo

## Introducción

- Conjunto de programas, imprescindible para el correcto funcionamiento del sistema, ya que controla tanto el ordenador como a los periféricos, y supervisa el funcionamiento de cualquier programa del usuario.
- El S.O. Está compuesto por diversos módulos, cada uno de los cuales tiene una función específica: leer datos del teclado, escribir en pantalla etc.
- Funciones del S.O:
  - Posibilita la utilización compartida de los recursos: discos, memoria, impresoras etc., impidiendo los conflictos.
  - Aumenta la eficacia global del ordenador.

*¿ Funciona el ordenador sin S.O?*

# Sistema Operativo

## Introducción

- Estructura del S.O típico.
  - Núcleo.
  - Gestor de Memoria.
  - Gestor de las operaciones de entrada/salida.
  - Gestión de los dispositivos de almacenamiento masivo.
  - Asignación de los recursos.
  - El Scheduler o planificador.

# Sistema Operativo

## Núcleo

- Modulo de más bajo nivel, actúa sobre el hardware.
- Funciones más importantes:
  - Conmutador de procesos o dispatcher (elige el siguiente proceso a ejecutar entre los que están esperando).
  - Comunicación entre programas y SO.

# Sistema Operativo

## Gestor de Memoria

- Módulo que se encarga de asignar zonas de la memoria principal a cada uno de los procesos que se están ejecutando.
  - Reparto transparente, eficiente y seguro.
  - La memoria es uno de los recursos más importantes de la computadora.

# Sistema Operativo

## Gestor de las operaciones de entrada/salida

- Los distintos dispositivos de entrada/salida tienen diferentes características y velocidades de acceso.
- Se trata de un abstracción lógica para el usuario y para los procesos.
- No es necesario conocer las características de los diferentes dispositivos.

# Sistema Operativo

## Gestión de los dispositivos de almacenamiento masivo

- Mantiene la estructura de toda la información y asegura un uso eficiente de los medios de almacenamiento masivo.
- Supervisa la creación, actualización y borrado de los archivos.



# Sistema Operativo

## Asignación de los recursos

- Optimización de la asignación de los recursos, proporcionando a cada proceso el que necesita, y en el momento en que le corresponde.
- Control de bloqueos entre procesos.
  - Interbloqueo: Varios procesos pueden necesitar un mismo recurso.
  - Inanición: Un proceso nunca recibe acceso a un recurso por baja prioridad y muere.

# Sistema Operativo

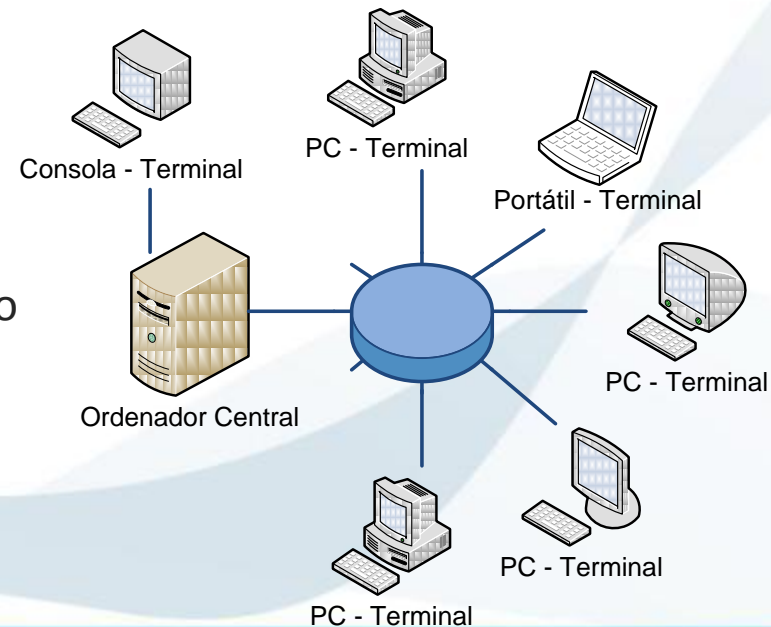
## El Scheduler o planificador.

- Asigna tiempos de procesador a los programas de acuerdo a una política de planificación que varía en cada SO.
  - Jerarquía de prioridades para asignar tiempo de CPU a cada proceso.

# Sistema Operativo

## Tipos de Sistema Operativo

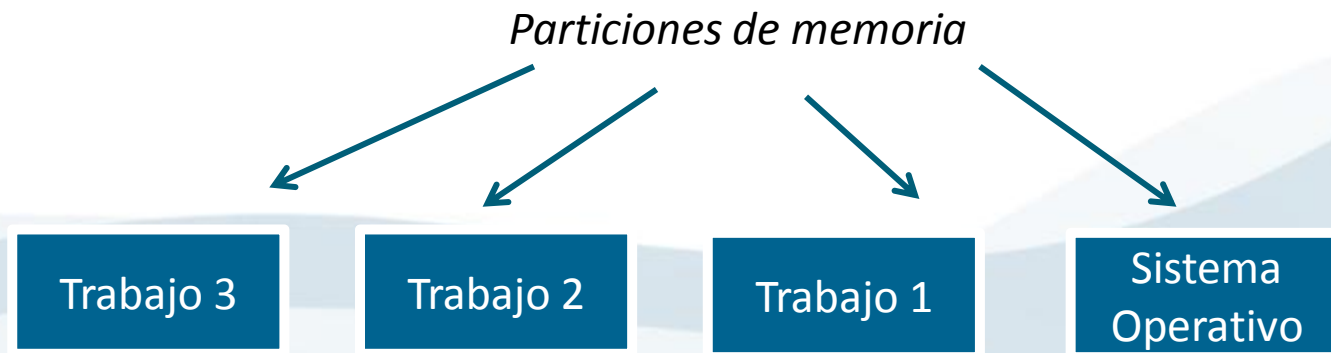
- **Monousuario.**
  - Orientados a equipos tipo PC.
    - ✓ Solo admite un usuario (una sola pantalla y un solo teclado).
  - Suelen disponer de una gran cantidad de programas de aplicación.
    - ✓ Ejemplos: MS-dos, versiones domésticas de Windows.
- **Multiusuario.**
  - Permiten trabajar en el mismo ordenador simultáneamente a diferentes usuarios.
  - Es habitual que cada usuario disponga de su propio perfil o configuración: entorno de trabajo (interfaz), acceso a determinados archivos.
    - ✓ Ejemplo: VMS, UNIX.



# Sistema Operativo

## Tipos de Sistema Operativo (II)

- Multiprogramación.
  - Obtener un rendimiento máximo de las distintas unidades que componen el ordenador. La velocidad de una CPU es superior a la de los periféricos.
  - Ocupación máxima de la CPU mediante canales (procesos, hebras).
  - La CPU realiza la ejecución de dos o más programas de forma concurrente con la finalidad de eliminar tiempo ocioso.



# Sistema Operativo

## Tipos de Sistema Operativo (III)

- Tiempo Compartido.
  - Utilización simultánea del sistema por varios usuarios y programas, como si el ordenador perteneciera exclusivamente a cada uno de ellos. Es decir, las respectivas tareas son resueltas por sus propios programas.
  - En tal sentido, es necesaria una multiprogramación, de tal manera que el ordenador atienda a los diferentes programas.
- Para Multiproceso.
  - Varios procesadores, funcionando simultáneamente y compartiendo memorias centrales y periféricas.
  - Se utiliza este sistema por razones de potencia o seguridad.
  - Se hace uso de la multiprogramación y del tiempo compartido.

# Sistema Operativo

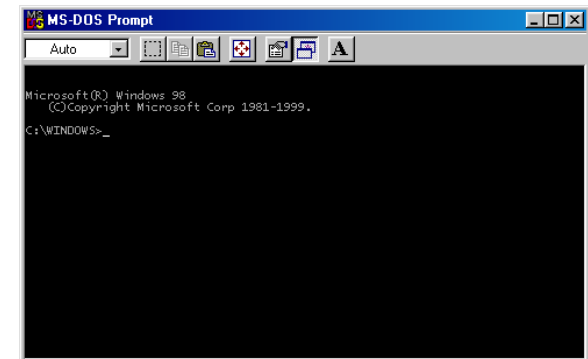
## Ejemplos de Sistema Operativo

- DOS (Disk Operating System), 1981.
  - Conjunto de programas cuya ejecución puede solicitar el usuario mediante órdenes sencillas a través de una consola de comandos en modo texto.
    - ✓ No tiene interfaz gráfica.
  - Las órdenes, también llamadas comandos, activan directamente los recursos del sistema a medida que el usuario lo solicita.
    - ✓ La mayoría sirven para gestionar los dispositivos de almacenamiento masivo (disquetes, discos duros, etc.).
  - No es multiusuario, ni multitarea.
  - Múltiples adaptaciones: MS-DOS, PC-DOS, DR-DOS, FreeDOS, PTS-DOS, ROM-DOS, etc-

# Sistema Operativo

## Ejemplos de Sistema Operativo (II)

- MS DOS (Microsoft Disk Operating System), 1981.
  - Versión de Microsoft del DOS para PCs.
    - ✓ La más popular y extendida.
  - Tuvo ocho versiones principales (hasta el año 2000), pero fue siendo reemplazado de forma gradual por S.O. que ofrecían una interfaz gráfica de usuario (GUI), en particular por varias generaciones del sistema operativo Microsoft Windows
    - ✓ A partir de Windows ME y Windows XP ya está completamente integrado.
  - Ejemplos de interacción:
    - ✓ Comandos:
      - Dir \*.\*
      - Delete «*fichero*»
      - Cd..
      - Cd «*ruta*»
      - Copy «origen» «destino»

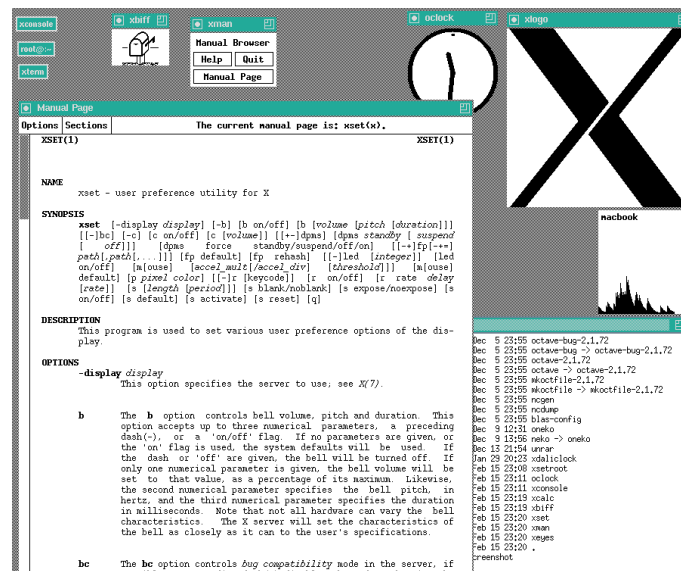


Ventana de MS-DOS o Símbolo del Sistema en Windows 98  
(Menú Inicio → Menú Ejecutar → «Cmd.exe»)

# Sistema Operativo

## Ejemplos de Sistema Operativo (III)

- Unix, 1969.
  - Creado por los laboratorios Bell de AT&T.
    - ✓ Entre otros Ken Thompson, Dennis Ritchie y Douglas McIlroy.
  - Multiusuario, Multitarea. Soporta multiprogramación.
  - Tiene varias versiones, para miniordenadores y microordenadores
  - Escrito en lenguaje C, dispone de un lenguaje ensamblador propio, varios compiladores y un editor de texto.
  - Varios usuarios acceden a un único procesador a través de terminales.
  - Interfaz gráfica mediante X Window System o Sistema de Ventanas X (aprox. 1985).



*Sistema de Ventanas para UNIX  
(finales década de los 80)*



# Sistema Operativo

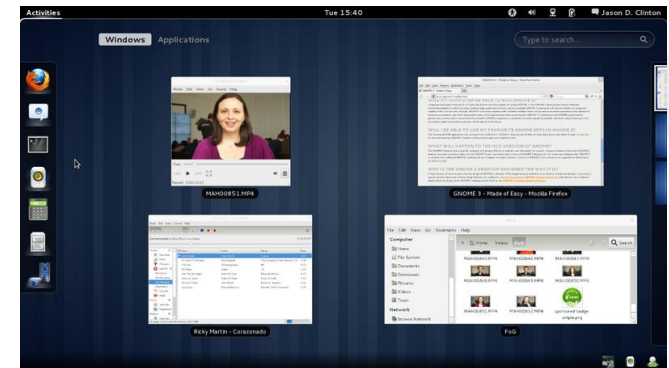
## Ejemplos de Sistema Operativo (IV)

- Xenix, 1979.
  - Creado por Microsoft a partir de una licencia de UNIX.
  - Intento adaptarlo para su uso en microprocesadores de 16 bits.
    - ✓ Fracasó y lo abandonó en 1986 para desarrollar OS/2 con IBM.
  - Era un S.O. multiusuario y multitarea.
- OS/2, 1987.
  - Desarrollado inicialmente de manera conjunta entre Microsoft e IBM para suceder a DOS.
    - ✓ Microsoft se centró en Windows 3.0.
  - Primer S.O. para máquinas de 32 bits (1992).

# Sistema Operativo

## Ejemplos de Sistema Operativo (V)

- Minix , 1987.
  - Creado por Andrew S. Tanenbaum.
  - Para enseñar a sus alumnos el diseño de sistemas operativos.
  - Clon de UNIX.
  - Disponible su código fuente (licencia BSD).
    - ✓ Permite reutilizar y modificar el programa.
- Linux, 1994.
  - El nombre proviene de su creador, el estudiante finlandés Linus Tordvals.
    - ✓ Dedico 4 años a la redacción de la primera versión para uso personal para no recurrir a MINIX.
  - Licencia de uso pública y gratuita (licencia GNU).
  - Multitarea y multiproceso.
  - Múltiples distribuciones (algunas comerciales): Debian, Ubuntu, Fedora, Open-SUSE, etc.



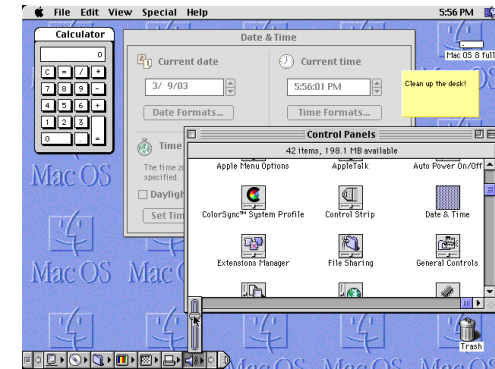
*Distribución Debian con  
interfaz Gnome 3.0*

Fuente Wikipedia

# Sistema Operativo

## Ejemplos de Sistema Operativo (VI)

- Mac OS (Macintosh Operating System), 1984.
  - Creado por Apple para su línea de computadoras Macintosh.
  - Primer SO dirigido al gran público en contar con una interfaz gráfica compuesta por la interacción del mouse con ventanas, iconos y menús.
  - Se mantiene hasta 1999 (versión 9.0), dando paso a Mac OS X basado en UNIX.



*Mac OS versión 8*

Fuente Flickr



*Mac OS X*

Fuente Flickr

# Sistema Operativo

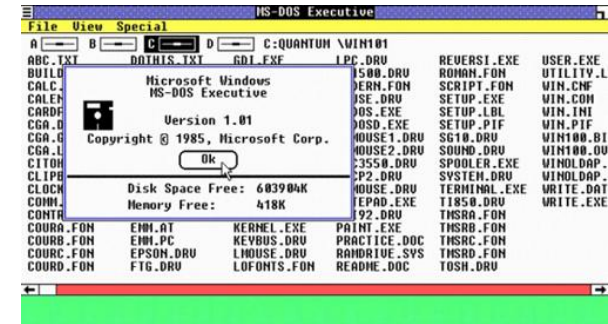
## Ejemplos de Sistema Operativo (VII)

- Microsoft Windows, 1985.
  - Conjunto de S.O. desarrollados por Microsoft desde 1985 (Windows 1.0) hasta 2009 (Windows 7).
  - En sus orígenes era la interfaz gráfica de MS-DOS .
    - ✓ Hasta Windows 95 no se trata de un S.O. completo.
  - Microsoft desarrolló su propia interfaz basada en ventanas.
    - ✓ Gran éxito comercial con Windows 3.0.
  - Aparecen multitud de paquetes de ofimática soportadas bajo el mismo S.O.
  - Dos familias: una para equipos sobremesa (familia Home) y otra para servidores (familia NT y Server).
    - ✓ A veces se mezclan como en XP Professional, Vista o Windows 7.

# Sistema Operativo

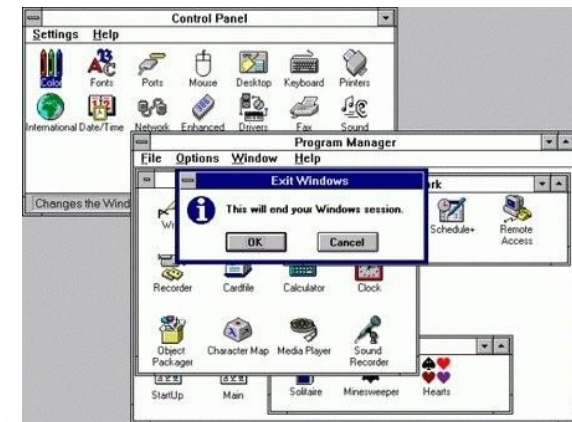
## Ejemplos de Sistema Operativo (VIII)

- Microsoft Windows. Características:
  - Dispone de una interfaz gráfica y sencilla.
  - Ofrece ayuda permanente.
  - Gestión de archivos y directorios sencilla.
  - Los S.O. para redes, Windows NT o Server, permite compartir todo tipo de recursos entre usuarios y grupos.
  - Los elementos básicos de la interfaz gráfica de Windows son: Escritorio, ventanas, iconos, barra de títulos, menús, etc.



## Windows 1.0

Fuente Flickr

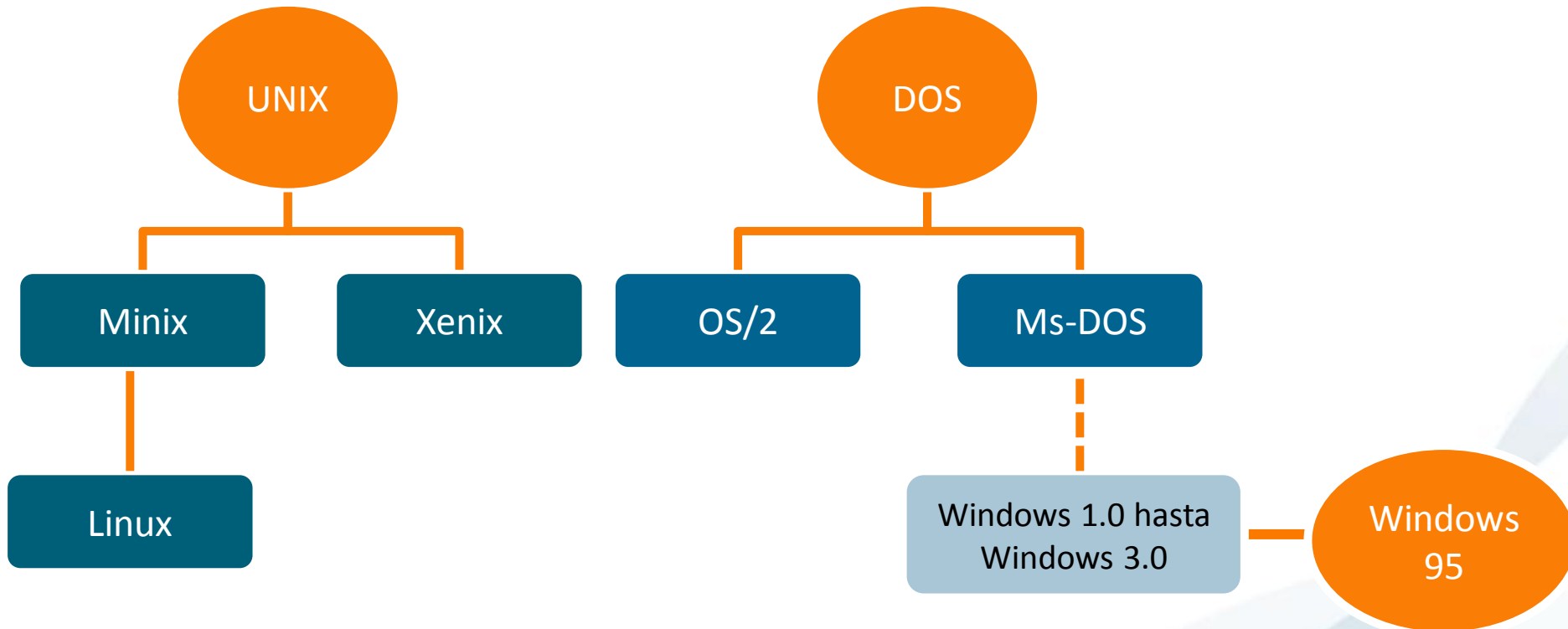


## Windows 3.11

Fuente Flickr

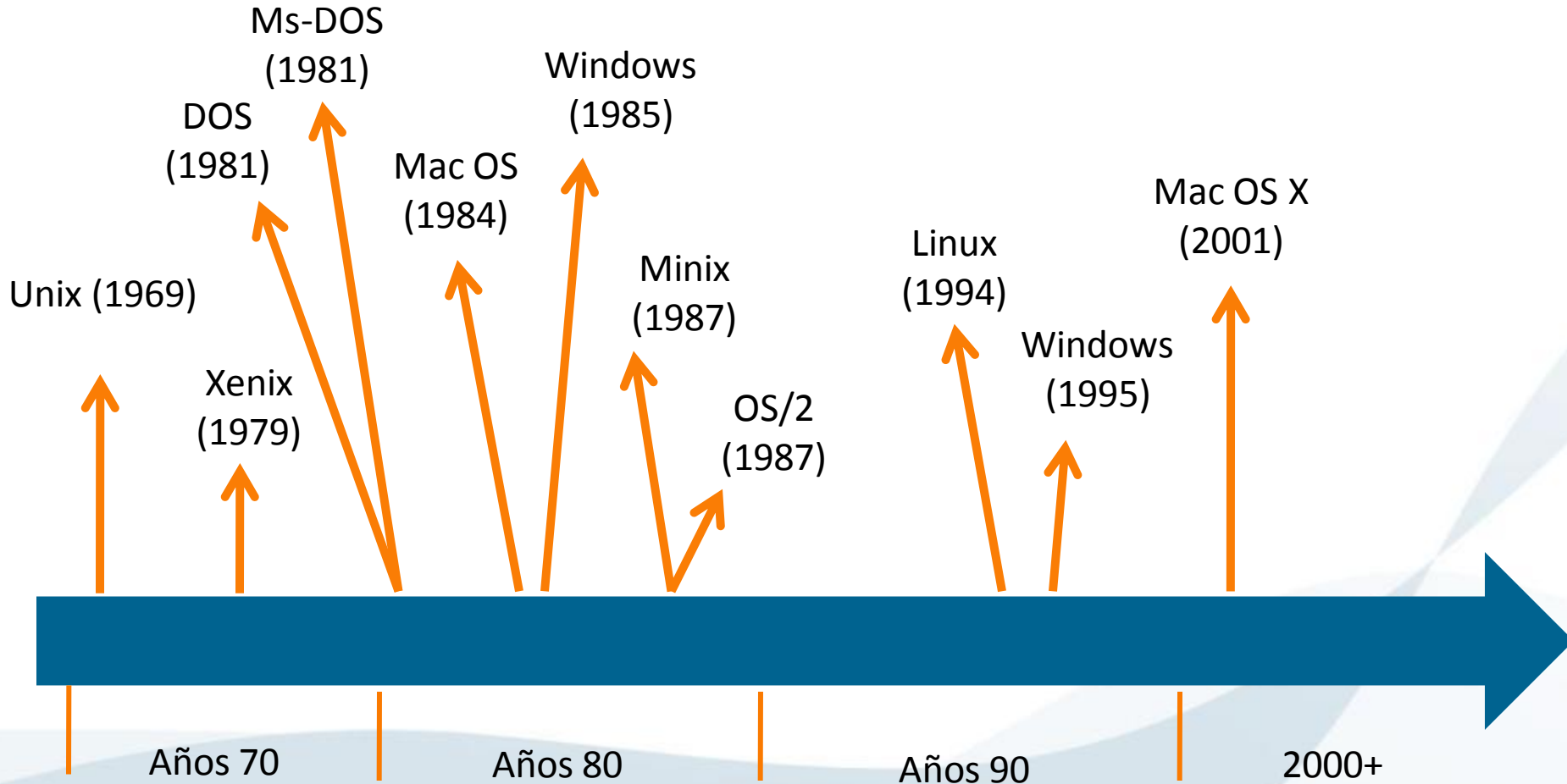
# Sistema Operativo

## Resumen de Sistemas Operativos. Jerarquía.



# Sistema Operativo

## Resumen de Sistemas Operativos. Evolución temporal.





# Lenguajes de Programación

## Índice

- Algoritmo.
- Elementos y características de los lenguajes de programación.
- Tipos de lenguajes de programación.
  - De bajo nivel.
  - De alto nivel.
  - Ejemplos de lenguaje de alto nivel.
  - Traductores, compiladores, intérpretes.



# Lenguajes de Programación

## Algoritmo

- Organigramas. Pseudocódigos.
  - Un algoritmo es la progresión exacta, detallada y sin ambigüedad de la secuencia de pasos elementales, para encontrar una solución. Y debe contemplar todas las alternativas posibles.
  - Método de resolución de un problema particular en un número finito de pasos.
  - No existe un lenguaje único y universal para la descripción de un algoritmo (lenguaje natural).
  - Un algoritmo no está vinculado a una máquina concreta, mientras que un programa si puede estarlo.
  - Un mismo problema puede tener varios métodos distintos de solución y en consecuencia distintos algoritmos.
  - Un algoritmo es la descripción de los pasos de una tarea.

# Lenguajes de Programación

## Algoritmo

- PseudoCódigo. Ejemplo.

**Leer un numero entre 1 y 100 y luego imprimir el cuadrado del número.**

Algoritmo Cuadrado

Repetir

    Escribir('Ingrese un número entre 1 y  
        100 para obtener el cuadrado del mismo')

    Leer(Numero)

Hasta Numero>=1 y Numero<=100

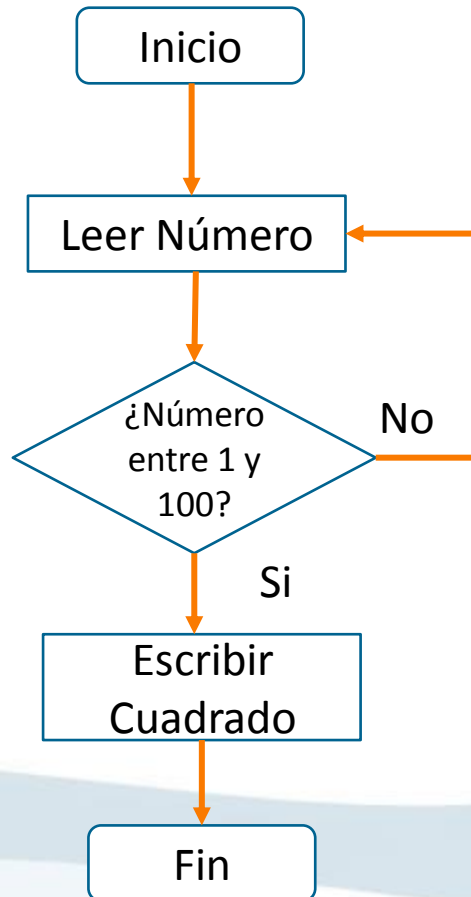
    Escribir ('El cuadrado es:', Numero \* Numero)

Fin Algoritmo

# Lenguajes de Programación

## Algoritmo

- PseudoCódigo. Ejemplo con Diagrama de Flujo.



# Lenguajes de Programación

## Elementos y características de los lenguajes de programación

- Los lenguajes de programación son herramientas que permiten escribir las instrucciones del programa, no a base de ceros y unos, sino mediante símbolos adecuados (letras, signos numéricos y especiales).
- Un lenguaje de programación posee y queda definido por un alfabeto, un vocabulario, ciertas reglas de sintaxis y una semántica.
  - Alfabeto: conjunto de símbolos empleados.
  - Vocabulario: conjunto de palabras formadas mediante símbolos, con significado definido.
  - Sintaxis : las reglas que permiten construir sentencias válidas.
  - Semántica: significado de las sentencias.

# Lenguajes de Programación

## Tipos de lenguajes de programación

- Lenguaje de bajo nivel.
  - Cierta analogía con las instrucciones de lenguaje máquina (0,1).
  - Código de operación tipo nemotécnico, lo que facilita la programación.
    - ✓ Por ejemplo SUM (sumar) o ADD (adición).
  - Direcciones simbólicas (se le asigna un nombre a cada dato y se referencia el nombre).

	Suma	celda1	celda2
Lenguaje máquina:	17	10125	10280 (todo en binario)
Lenguaje simbólico:	SUM	cant1	cant2

- Uso de lenguajes ensambladores, que son diferentes para cada ordenador.

# Lenguajes de Programación

## Tipos de lenguajes de programación (II)

- Lenguaje de alto nivel o evolucionarios.
  - Permite escribir instrucciones orientadas al problema que se desea resolver.
  - Terminología fácilmente comprensible y que se aproxima más o menos al propio lenguaje humano.
  - Una instrucción se traduce en varias instrucciones máquina, lo contrario de un lenguaje de bajo nivel.
  - Instrucciones compartidas por todos los lenguajes son:
    - ✓ Movimiento de datos (mover, guardar).
    - ✓ Asignación de datos.
    - ✓ Operaciones aritméticas (sumar, restar, etc.).
    - ✓ Operaciones lógicas (AND, OR, NOT, XOR).
    - ✓ Transferencias de control ( bucles, subrutinas).
    - ✓ Instrucciones de entrada y salida.

# Lenguajes de Programación

## Tipos de lenguajes de programación (III)

- Lenguaje de alto nivel o evolucionarios.
  - Ventajas:
    - ✓ Facilidad de aprendizaje.
    - ✓ Sencillez de utilización.
    - ✓ Potencia de las instrucciones.
  - Desventajas:
    - ✓ La utilización de estos lenguajes puede obligar a emplear unos elementos de hardware adicionales o ampliados.

# Lenguajes de Programación

## Tipos de lenguajes de programación (IV)

- Lenguaje de alto nivel o evolucionarios.
  - Ejemplos:
    - ✓ FORTRAN (Formula Translation).
      - Creado por John Backus (IBM) en 1956.
      - Lenguaje de propósito general diseñado para aplicaciones científicas y técnicas basado en fórmulas matemáticas.
      - Facilita el uso de matrices y maneja con dificultad datos no numéricos.
    - ✓ COBOL (Common Business Oriented Language).
      - Surge entre 1957 y 1960 a partir del trabajo de Grace Murray Hopper.
      - Lenguaje estándar para programar ordenadores que procesan datos comerciales que normalmente tratan gran cantidad de datos pero sin someterlos a cálculos complejos.
    - ✓ BASIC (Beginner's All Purpose Symbolic Instruction Code).
      - Apareció en 1964 creado por Kurtz y Kennedy.
      - Se diseñó para introducir estudiantes de ciencia en la programación.



# Lenguajes de Programación

## Tipos de lenguajes de programación (V)

- Lenguaje de alto nivel o evolucionarios.
  - Más ejemplos:
    - ✓ PASCAL.
      - Creado por Niklaus Wirth en 1970
      - Objetivo principal es el aprendizaje de estructuras concisas para su uso en diferentes metodologías de programación.
    - ✓ LENGUAJE C.
      - Creado por Dennis Ritchie en 1972.
      - El Sistema Operativo UNIX está escrito en C.
      - Se usa fundamentalmente para diseñar software de sistemas comerciales y científicos

# Lenguajes de Programación

## Tipos de lenguajes de programación (VI)

- Lenguaje de alto nivel o evolucionarios.
  - Más ejemplos:
    - ✓ ADA.
      - 1980, desarrollado por el Ministerio de Defensa de los EEUU
      - Incluye la multitarea.
      - Toma el nombre de Ada Lovelace, primera persona en escribir un programa para una computadora programable.
    - ✓ JAVA.
      - Desarrollado por Sun Microsystems a principios de los años 90.
      - Parte de su sintaxis está basada en C y C++.
      - Java surgió para contrarrestar el avance de Microsoft en el software relacionado con Internet
      - La novedad radica en la incorporación de componentes que permiten su uso en cualquier tipo de plataformas
      - Java es en la actualidad el lenguaje de programación más utilizado en aplicaciones relacionadas con la Web.

# Lenguajes de Programación

## Tipos de lenguajes de programación (VII)

- Generaciones bajo el punto de vista de los lenguajes empleados.
  - Primera Generación : Lenguaje máquina.
  - Segunda Generación: Lenguaje ensamblador.
  - Tercera Generación: Lenguaje de alto nivel.
    - ✓ Ejemplos: Fortran, Cobol, Basic, Pascal, Lenguaje C.
  - Cuarta Generación: Lenguajes orientados al usuario.
    - ✓ Ejemplos: Visual Basic, Visual C.
  - Quinta Generación: Lenguajes naturales.
    - ✓ Ejemplo: Prolog.

# Lenguajes de Programación

## Traductores

- Los traductores utilizan como datos de entrada las instrucciones de un programa escritas en caracteres inteligibles o simbólicos (llamado también programa fuente), dan como resultado el programa escrito en lenguaje máquina (llamado también programa objeto).
- El traductor verifica las reglas del lenguaje que utiliza.
- Tipos de traductores:
  - Ensambladores
  - Compiladores
  - Intérpretes
  - Preprocesadores
  - Compiladores cruzados

# Lenguajes de Programación

## Traductores (II)

- **Ensambladores**

- Se emplean para traducir lenguajes ensambladores. La traducción se hace instrucción a instrucción, cada instrucción del lenguaje ensamblador da lugar a una instrucción en lenguaje maquina.

- **Compiladores**

- La traducción se hace sobre programas escritos en lenguaje de alto nivel.
- Existen compiladores como tantos lenguajes de alto nivel.
- El proceso de traducción es ejecutado una sola vez y el programa objeto obtenido se puede guardar indefinidamente para ejecutarlo cuantas veces se requiera.

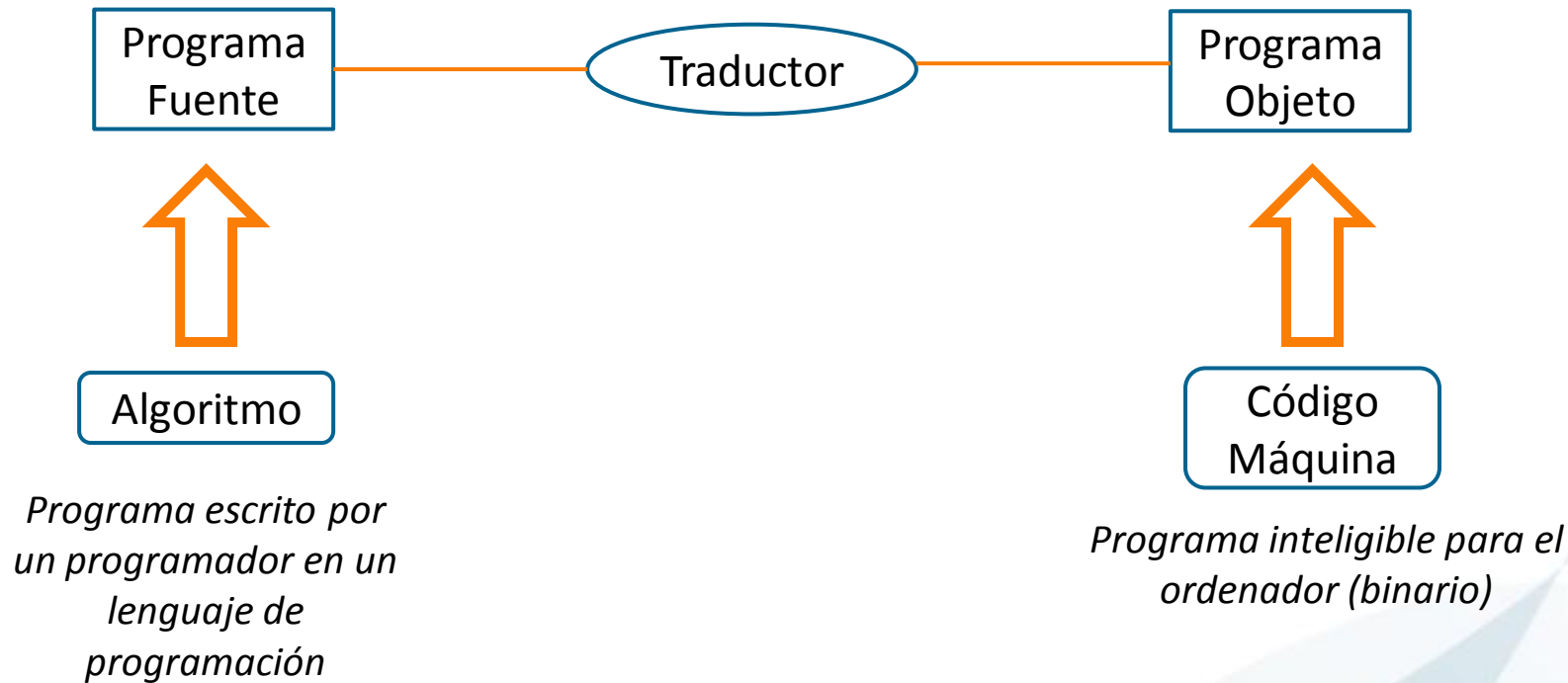
# Lenguajes de Programación

## Traductores (III)

- **Intérpretes**
  - Algunos lenguajes, como el BASIC, admiten la posibilidad de ser interpretados en vez de compilados.
  - El interprete no genera ningún programa objeto.
  - Se suele utilizar en la fase de diseño y desarrollo de un programa, debido a la comodidad de ir detectando los errores instrucción por instrucción, en vez de tener que esperar a traducir todo y luego ir descubriendo los errores.
  - Una vez que se comprueba que el programa no tiene errores se procede a compilar y obtener el programa-objeto, que será guardado y ejecutado cuantas veces sea necesario.
- **Preprocesadores**
  - Partiendo de un programa escrito en un lenguaje de alto nivel obtienen un fuente de otro lenguaje del mismo nivel.
- **Compiladores cruzados**
  - Son compiladores que transforman un programa fuente en un programa objeto de otra maquina distinta de aquella sobre la que trabaja el compilador cruzado.

# Lenguajes de Programación

## Traductores (IV)



# Ingeniería del Software



*Importancia de la información en una empresa*



# Ingeniería del Software (II)

1. Informatización de la administración y de la gestión empresarial.
2. Análisis, Diseño, Programación y Documentación de un sistema informático.
3. Ciclo de vida de una aplicación informática.

# Ingeniería del Software

## I. Informatización de la administración y de la gestión empresarial

- Software : es la suma total de los programas de los ordenadores, procedimientos, reglas que pertenecen a un sistema.
- La Ingeniería del Software es la disciplina que trata del análisis, diseño y programación de un sistema informático.
  - Permite desarrollar sistemas siguiendo una serie de fases o etapas que constituyen el Ciclo de vida de un sistema.

# Ingeniería del Software

## I. Informatización de la administración y de la gestión empresarial (II)

- Objetivos
  - Facilitar el tratamiento de grandes volúmenes de datos
  - Reducir el coste de la administración y de la gestión empresarial
  - Aumentar la fiabilidad, seguridad y prontitud de los resultados
  - Integrar entre sí varios sistemas informáticos
  - Rentabilizar el cambio de gestión manual a la informatizada

# Ingeniería del Software

## I. Informatización de la administración y de la gestión empresarial (III)

- La informatización de un sistema de información, debe cumplir los siguientes requisitos:
  - Los resultados serán los previstos en los objetivos
  - Los costes no excederán del máximo previsto
  - El tiempo de desarrollo estará dentro de los límites prefijados
  - El sistema debe documentarse debidamente en cada una de las etapas de la realización del software.

# Ingeniería del Software

## I. Informatización de la administración y de la gestión empresarial (IV)

- Incorporación de una Metodología de desarrollo software para mejorar la eficiencia:
  - Conjunto de prácticas recomendadas (métodos y técnicas), acompañadas a menudo de materiales de aprendizaje y herramientas.
  - Ejemplos: Métrica 3, Metodología ESA

# Ingeniería del Software

## I. Informatización de la administración y de la gestión empresarial (V)

- Problemas comunes (a evitar) en el desarrollo de Software:
  - Comunicación deficiente.
    - ✓ Con el cliente.
    - ✓ Con el equipo de desarrollo.
  - Se termina el software fuera de los plazos y precios establecidos.
  - Cambios de los requisitos por parte del cliente.
  - Ausencia de documentación.
  - Errores en todas las fases del desarrollo.
  - Incremento de costes.

# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático

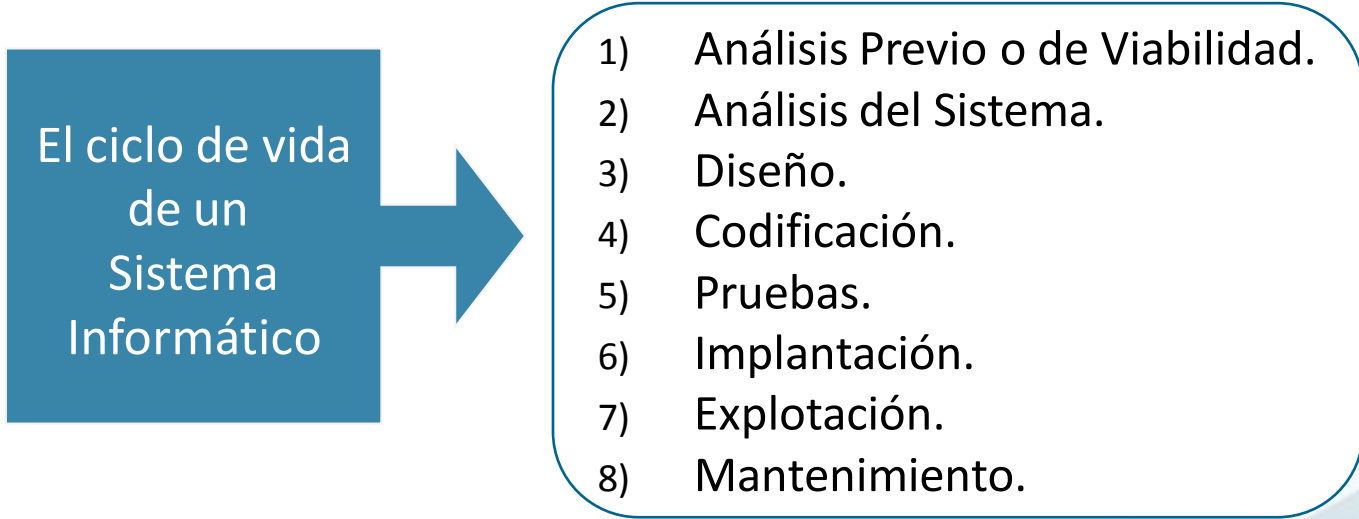
- El ciclo de vida de un sistema informático comprende el conjunto de fases que transcurren desde que se detecta la necesidad de informatizar un proceso empresarial, hasta que sea implantado en la empresa.
- Cada una de las fases genera informes que se reúnen y se estructura para documentar el sistema.

# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático (II)

- Casi todas las metodologías manejan como conceptos las siguientes fases del ciclo de vida:

El ciclo de vida  
de un  
Sistema  
Informático



- 1) Análisis Previo o de Viabilidad.
- 2) Análisis del Sistema.
- 3) Diseño.
- 4) Codificación.
- 5) Pruebas.
- 6) Implantación.
- 7) Explotación.
- 8) Mantenimiento.



# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático (III)

1. Análisis Previo  
o de Viabilidad



Viabilidad técnica y económica.

2. Análisis del  
Sistema



Determinación de los requisitos  
(comunicación con el cliente):

- Elicitar.
- Analizar.
- Validar.

# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático (IV)

3. Diseño



Determinación de lo que se va a codificar.

4. Codificación



Se transforman en programas lo diseñado utilizando un lenguaje de programación.

# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático (V)

5. Pruebas



Verificación del correcto funcionamiento del sistema, generando el mayor número de casos que se puedan presentar en el sistema.

6. Implantación



Instalación del sistema y formación de usuarios.

# Ingeniería del Software

## II. Análisis, Diseño, Programación y Documentación de un sistema Informático (VI)

### 7.Explotación



- Fase de utilización del sistema desarrollado.
- Automatización de la administración y de la gestión en la forma mecanizada prevista.
- Copias de seguridad, controles, preparación de listados, etc.

### 8.Mantenimiento



- Garantizar el correcto funcionamiento del sistema durante el tiempo que dure su explotación.
- Incorporar las posibles modificaciones .

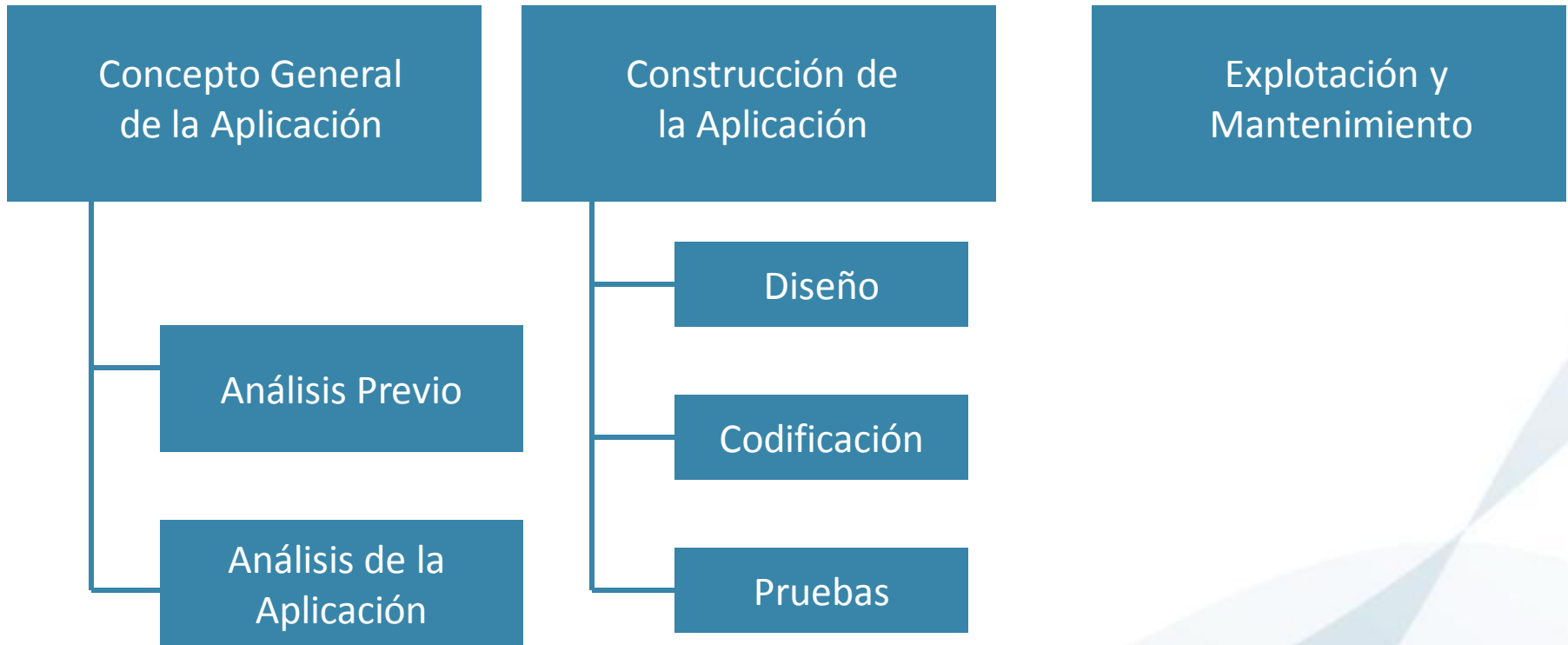
# Ingeniería del Software

## III. Ciclo de vida de una aplicación

- Una aplicación informática engloba un conjunto de tareas relacionadas que se desarrollan en el campo de la empresa.
  - Ejemplos : Gestión contable, gestión de Recursos humanos etc.
- La Ingeniería del software propone diseños orientados a las aplicaciones como partes constitutivas de un sistema informático.

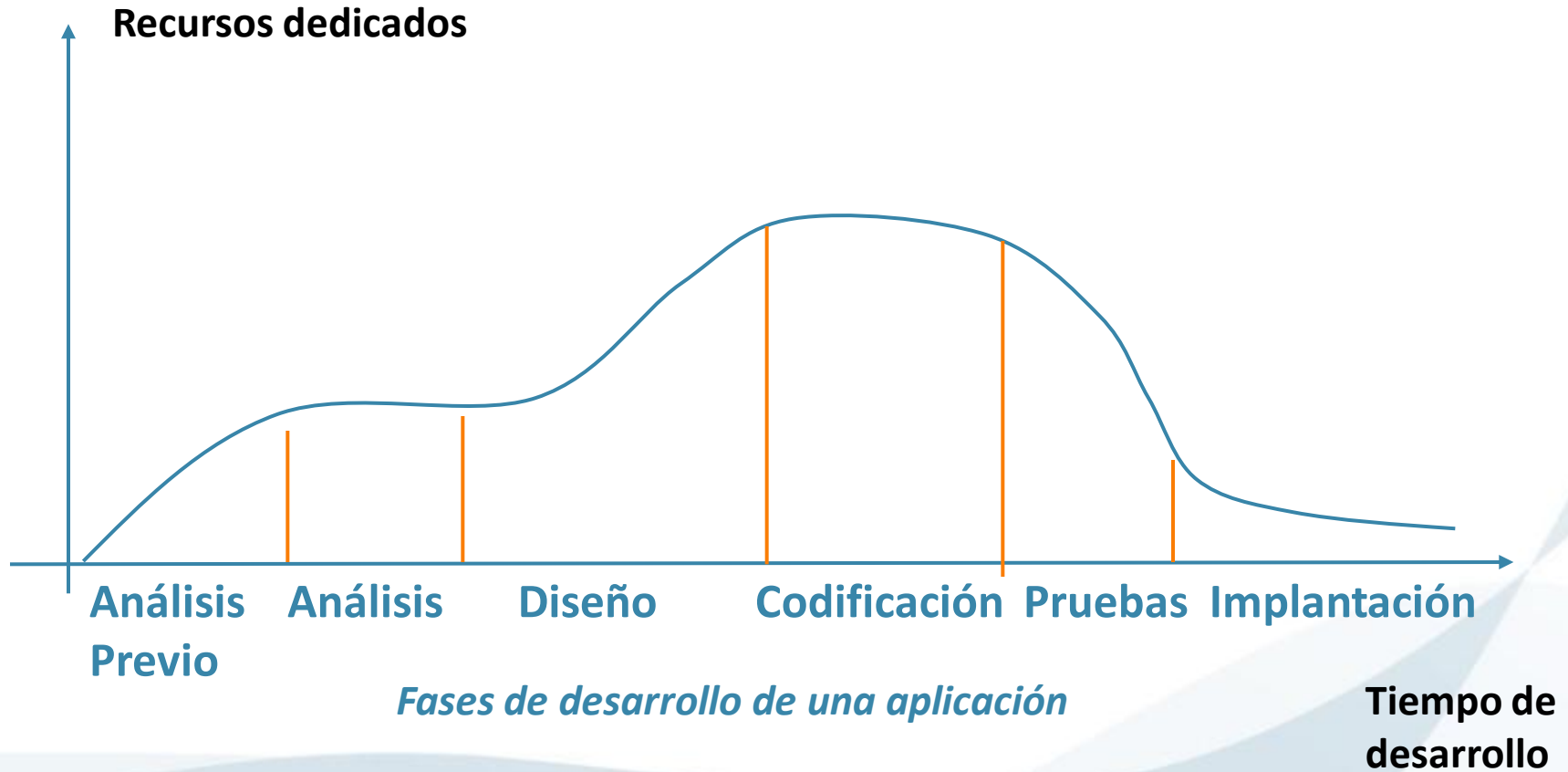
# Ingeniería del Software

## III. Ciclo de vida de una aplicación (II)



# Ingeniería del Software

## III. Ciclo de vida de una aplicación (III)



# Enfermedades del Software


## Virus

- Son una de las principales amenazas de las redes informáticas, es un programa que infecta ficheros ejecutables u objetos.
- Son programas que se pegan a un archivo que les facilita el acceso a la memoria del sistema o al SO.
- Pueden llegar a infectar el sector de arranque de un ordenador o de cualquier disquete al que tengan acceso.
- La mayoría siguen el mismo método para infectar un sistema.



# Enfermedades del Software

## Virus. Funcionamiento

- 
- Carga de un archivo infectado en la memoria del ordenador. Puede provenir de un disquete infectado, de la red local, o de internet.
  - Cuando se ejecuta, hace copias de sí mismo .
  - Una vez copiado en memoria, espera a que el usuario ejecute programas.
  - Al ejecutarse los programas en la memoria, el virus se pega a ellos.
  - Cuando se cierra el sistema, el virus desaparece de la memoria, pero no así la copia que se ha pegado a los programas.
  - Al ejecutar , otra vez, un programa infectado vuelve a realizarse todo el proceso anterior.

# Enfermedades del Software

## Virus. Tipos

- Por el modo que actúan infectando:
  - Virus de macro.
    - ✓ Infectan ficheros que se han creado con ciertas aplicaciones o programas ofimáticos
    - ✓ Hojas de calculo, documentos, bases de datos, etc.
    - ✓ Viajan con el archivo.
  - Virus de enlace o de directorio.
    - ✓ Afecta a la ubicación de los ficheros dentro de un disco duro (sistema de directorios del SO).
    - ✓ Suplanta la ruta original de un fichero por la del virus.
    - ✓ Al acceder al fichero en realidad se accede al virus.

# Enfermedades del Software

## Virus. Tipos (II)

- Por el modo que actúan infectando:
  - Virus de fichero.
    - ✓ Infecta programas y ficheros ejecutables (.exe o .com).
  - Virus de boot (sector de arranque).
    - ✓ Se activan en el momento en que se arranca el ordenador desde un disco infectado, ya sea desde el disco duro o desde un disquete.
    - ✓ Se propagan a la memoria del ordenador (RAM).

# Enfermedades del Software

## Virus. Tipos (III)

- Tipos más extendidos:
  - Caballos de Troya.
  - Polimorfos (cifrados).
  - Sigilosos.
  - Lentos.
  - Retro-virus.
  - Blindados.
  - Voraces.
  - Horax, etc.

# Enfermedades del Software

## Virus. Tipos (IV)

- Horax y Phishing Bancario.
  - Difundir a través de redes de ordenadores, rumores infundados que invitan a realizar tareas determinadas a los usuarios.
    - ✓ Se puede definir como una falsa alarma sobre un virus informático que se distribuye en cadena de mensajes por correo electrónico.
    - ✓ Estos mensajes deliberadamente falsos, son creados con la grave intención de provocar pánico.
    - ✓ Los usuarios ingenuos, caen en la trampa y siguiendo las instrucciones, empiezan a re-transmitirlos, pensando que de este modo van a ayudar a otros usuarios.
    - ✓ Estos hechos provocan una reacción en cadena que además del amedrentamiento, ocasionan la saturación de los buzones de correo y la consiguiente congestión de las conexiones en Internet.
  - Detección:
    - ✓ Dirección Web incorrecta.
    - ✓ Faltas de ortografía y/o mala traducción.
    - ✓ Puedes recibirlo de alguien de confianza que te lo reenvía.

# Enfermedades del Software

## Virus. Tipos (V)

- Horax y Phishing Bancario. Ejemplo.

### Estimado Cliente,

Nosotros hemos determinado eso fue 3 tentativas equivocadas a la entrada en su cuenta bancaria en línea del hostname: **21.38.77.82.static.cluj.rdsnet.ro**. Sospechamos que esta tentativa no fue legitimada así, como un measure de seguridad, nosotros hemos suspendido temporalmente su cuenta. Usted puede reactivar su cuenta, el tiempo que usted desea, verificando sus informaciones personales conectadas a su cuenta bancaria en línea.

Para reactivar su cuenta utiliza por favor [chasque aquí](#).

### Favor de notar:

**Su cuenta se quedará suspendió para prevenir el fraude hasta que usted lo reactivará.**

Gracias por utilizar servicios en línea Bankinter.  
El Equipo Bankinter

### PROTEGE SU CONTRASEÑA

Nunca dele su contraseña a nadie, inclusive empleados Bankinter.

Por favor no conteste a este correo electrónico. Este buzón no es vigilado y usted no recibirá una respuesta.

© Bankinter,S.A. Todos los derechos reservados.

# Enfermedades del Software

## Virus. Tipos (VII)

- **Gusanos.**
  - Es un programa informático que se reproduce a sí mismo en otras zonas de la memoria del ordenador diferentes a la que ocupa hasta que desborda la capacidad de almacenamiento de la memoria.
- **Bombas Lógicas.**
  - Es un programa nocivo que se oculta, ocupando una pequeña parte de la memoria y que actúa en una determina fecha.
- **Trojanos.**
  - Los caballos de Troya o trojanos son programas que se introducen en el ordenador por diversos medios, se instalan de forma permanente y tratan de tomar el control del sistema afectado

# Seguridad Informática

---

- Seguridad Informática
- Características a proteger.
- Riesgo
- Medidas de protección y seguridad



# Seguridad Informática

## Definición

- Son las técnicas desarrolladas para proteger programas o equipos informáticos individuales y conectados en una red frente a daños accidentales o intencionados
- En Informática, la seguridad debe proteger:
  - Propiedad intelectual
  - Confidencialidad : Impedir el conocimiento de algo por usuarios no autorizados.
  - Integridad: Impedir la modificación o eliminación, total o parcial, por usuarios no autorizados.
  - Disponibilidad: Garantizar la accesibilidad para los usuarios autorizados.
  - No-repudio: Registrar el autor de una acción de forma que éste no pueda negar haberla realizado (firma electrónica).

# Seguridad Informática

## Riesgo

- La medida del riesgo al que nos exponemos depende de:
  - Las amenazas : Si nadie nos amenaza, no hay riesgo.
  - La vulnerabilidad: Si nuestro sistema no es vulnerable, no hay riesgo.
  - El impacto: Si el impacto de los ataques es nulo, no hay riesgo.
- Tipos de Riesgo:
  - El propio usuario: causa en muchas ocasiones, errores y fallos no intencionados, debido a una mala manipulación del sistema. Suelen producirse por desconocimiento del funcionamiento del ordenador o por descuidos accidentales.
  - Programas maliciosos: destinados a perjudicar o a hacer uso ilícito de los recursos del sistema, abriendo una puerta a intrusos o bien modificando los datos del mismo..

# Seguridad Informática

## Riesgo (II)

- Intruso: persona que consigue acceder a los datos o programas de los cuales no tiene acceso permitido. Ejemplo: Hacker, Cracker, Spoofing.
- Siniestro: suele provocarse por desastres naturales, robo, incendio, corte de suministro eléctrico.

# Seguridad Informática

## Medidas de protección y seguridad

- Legales.
  - LPI: Ley de Propiedad Intelectual
  - LOPD: Ley Orgánica de Protección de datos,
  - LFE: La Firma Electrónica
- Administrativas, organizativas.
  - Plan Director de Seguridad.
- Físicas.
  - Contra incendios, inundaciones, apagones, etc.
- Software.
  - Política de respaldo o copias de seguridad.
  - Programas antivirus.
    - ✓ Son programas cuya función es detectar y eliminar virus informáticos y otros programas maliciosos

# Seguridad Informática

## Medidas de protección y seguridad (II)

- Técnicas:
  - Identificación y autenticación (clave y contraseña).
    - ✓ Elegir una contraseña fácil de recordar.
    - ✓ Que no esté en los diccionarios.
    - ✓ Utilizar un algoritmo que aporte números y otros cambios.  
Caracteres parecidos y letras que se parecen a los números.
  - Control y limitación de acceso a los datos y recursos.
  - Criptografía: sistemas que ofrecen medios seguros de comunicación en los que el emisor oculta o cifra el mensaje antes de transmitirlo para que sólo un receptor autorizado pueda descifrarlo.

# Seguridad Informática

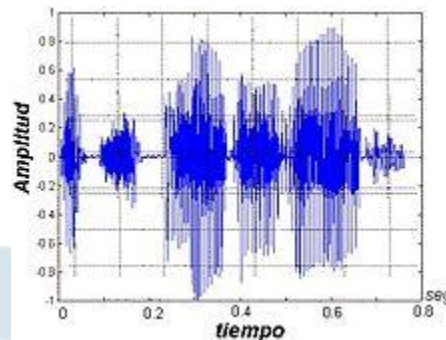
## Medidas de protección y seguridad (III)

- Técnicas:
  - Autenticación biométrica: Tecnologías para medir y analizar las características físicas y del comportamiento humanas con propósito de autenticación.
    - ✓ Huellas dactilares.
    - ✓ Mapa de la venas de la mano.
    - ✓ La onda sonora de la voz.
    - ✓ Iris



Huella  
Dactilar

Fuente Wikipedia



Fuente Wikipedia

Señal de  
la voz



Iris

Fuente Flickr