

# Introducción a los sistemas operativos

Gunnar Wolf



# Índice

- 1 ¿Qué son y qué hacen?
- 2 Historia y evolución
- 3 Computadoras personales
- 4 Dispositivos móviles



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora
- El programa que *siempre corre*



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora
- El programa que *siempre corre*
- Gestor de los *recursos* del sistema



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora
- El programa que *siempre corre*
- Gestor de los *recursos* del sistema
- Lo que define *qué es compatible* y *qué no* dentro de una determinada arquitectura



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora
- El programa que *siempre corre*
- Gestor de los *recursos* del sistema
- Lo que define *qué es compatible* y *qué no* dentro de una determinada arquitectura
- El programa *menos importante* de todos



# ¿Qué es un sistema operativo?

- El *sistema base* de una computadora
- El programa que *siempre corre*
- Gestor de los *recursos* del sistema
- Lo que define *qué es compatible* y *qué no* dentro de una determinada arquitectura
- El programa *menos importante* de todos
  - No realiza *trabajo útil*, sino que permite que *otros* lo hagan
- ...



## ... ¿Qué *no* es?

- Los programas básicos para administrar archivos
- La forma en que el usuario *lanza* programas
- El ambiente con que interactúa el usuario
  - Entorno gráfico
  - Línea de comando
  - ...



# ¿Qué brinda al *programador*?

- Abstracción
  - *Embelllecimiento* -Finkel
  - *Virtualización* -Arpaci-Dusseau
- Gestión de recursos
- Aislamiento y protección



# Abstracción

Proporciona *abstracciones consistentes y simplificaciones* a los procesos del usuario

- Archivos y directorios
- *Flujos de caracteres* (entrada/salida)
- Dispositivos, conexiones de red, contacto con el *mundo exterior*
- El concepto mismo de *proceso*



## Gestión de recursos

Administra los *recursos existentes* en la computadora, permitiendo la ejecución a los diversos procesos

- Cómo comparten los diversos procesos los recursos existentes (y rivales)
- Políticas de asignación (y recuperación) justas



# Aislamiento y protección

Protección de los datos, de los recursos, de los procesos

- Entre procesos
- Entre usuarios
- Ante procesos *mal comportados*
- Ante procesos maliciosos



# Ahora sí... ¿Qué es un sistema operativo?

- El principal programa que corre en una computadora de propósito general
- Provee una serie de abstracciones básicas a los programas
  - Pueden haber diferentes sistemas operativos, definiendo distintas *interfaces*, sobre el mismo hardware
  - Un mismo sistema operativo puede *adaptarse* a distintas arquitecturas de hardware
- Ofrece una infraestructura de gestión, aislamiento y protección de recursos
- Permite la implementación de *entornos operativos*



# Índice

- 1 ¿Qué son y qué hacen?
- 2 Historia y evolución**
- 3 Computadoras personales
- 4 Dispositivos móviles



# Construyendo a través de la historia

Para comprender lo que hoy gestionan los sistemas operativos, comencemos viendo cómo es que llegaron a gestionarlo.

Vamos con un repaso histórico de la historia de la computación, enfocados a las *innovaciones* de cada etapa



## En el principio...

- Arquitectura *von Neumann* (programa almacenado)
- Programación directa y explícita para el hardware
- Tiempo de programación → tiempo *no productivo* → desperdicio de recursos



## Sistemas de proceso por lotes (*batch processing*)

- Los programadores codifican su código en un medio de almacenamiento (*tarjetas perforadas*)
- Entregan los tambaches (*batches*) a los operadores
- Los operadores cargan secuencialmente los trabajos, entregan los resultados conforme se presentan



# Sistemas monitor en el proceso por lotes

- Implementan protección evitando la corrupción de *otros trabajos*
  - Interactuar con el lector de tarjetas (corrompiendo el siguiente programa)
  - Temporizadores y alarmas para evitar ciclos infinitos
- Estas medidas de protección requieren *modificación del hardware*
  - Noción de *instrucciones privilegiadas*



## Sistemas en lotes con *spool*

- ¿Spool? Bobina, o *Simultaneous Peripheral Operations On-Line*
- Cintas magnéticas
  - Carga intermedia de tarjeta a cinta
  - Resultados a cinta para posterior impresión
- Liberando los trabajos más lentos
  - Empleo de equipos *periféricos* especializados



# Sistemas multiprogramados

- Diferentes etapas en la vida de un proceso: *limitado por CPU, limitado por entrada-salida*
- Para maximizar el uso de recursos, ejecutar *simultáneamente* varios procesos
  - Requiere cambios fuertes en el hardware
  - Protección de recursos — Espacio de memoria
  - Recursos *estrictamente secuenciales* requieren *bloqueos* para ofrecer *acceso exclusivo*
- El monitor es invocado con mucha mayor frecuencia por los temporizadores
  - *Cambios de contexto*
- Interacción con el equipo: Se mantiene como el modelo en lotes



# Sistemas de tiempo compartido

- 1960s: Sistemas *interactivos y multiusuarios*
- Manejo de *terminales* para la interacción (teletipos, CRTs)
- Abstracciones de almacenamiento: Archivos, directorios en discos
- Ventajas al programador:
  - Interacción directa con el equipo
  - Edición interactiva
  - Compilación parcial
  - Ejecución inmediata
  - Bibliotecas de sistema
- Complejidad técnica
  - Requisito de múltiples cambios de contexto por segundo



# Tipos de multitarea

*Cooperativa o no apropiativa* (cooperative multitasking) Cada proceso tiene control del CPU hasta que efectúa una *llamada al sistema* o indica al sistema que puede tomar el control (`yield`)

*Preventiva o apropiativa* (preemptive multitasking) El reloj del sistema interrumpe la ejecución de cada proceso transfiriendo *forzosamente* el control al sistema operativo



# Clases de procesos

¿Qué procesos son más importantes?

**Procesos interactivos** El usuario tiene la *experiencia* de la demora

**Del sistema** Procesos no postergables

**Por nivel de usuario** Hay usuarios más importantes que otros

**... Asegún el sapo** Cada situación puede ameritar una política diferente

Los procesos se organizan en *colas de prioridad* según las políticas requeridas por cada sistema



# Índice

- 1 ¿Qué son y qué hacen?
- 2 Historia y evolución
- 3 **Computadoras personales**
- 4 Dispositivos móviles



## Nacimiento de las computadoras personales

- En los 1970s comienzan a aparecer las computadoras personales
- En un principio, programadas a través de *switches*, con resultados a través de LEDs



Figura: Microcomputadora Altair 8800 (1975,  $\approx$  US\$600)



## La era de los 8 bits ( $\approx$ 1977 – 1985)

- *Microprocesadores* de 8 bits y miniaturización
  - Salida a video (tipo TV)
  - Entrada por teclado
  - Entrada opcional por cinta, primeros *diskettes* (discos flexibles)
- Programación en BASIC (intérprete en ROM)



Figura: Commodore Pet  
2001 (1977)



## La era de los 8 bits ( $\approx$ 1977 – 1985)

- Comienzan a manejarse *dispositivos*: Unidades de cinta, unidades de disco, impresoras, modems, etc.
- Muchas arquitecturas mutuamente incompatibles
- Separan el *entorno de desarrollo* del *entorno de ejecución*
  - Nace la *distribución de binarios*
- Explosión de la industria de los videojuegos



## La microcomputadora *seria*: Familia PC (1981)



**Figura:** Computadora IBM PC modelo 5150 (1981)

- Primer computadora de una empresa *seria*, orientada a su uso en ambiente empresarial
  - Sin color ni audio... ¿Para qué?
- Entorno primario de ejecución: *Línea de comando* (PC-DOS, MS-DOS)
- Al día de hoy sigue siendo la arquitectura predominante



## Entorno gráfico (WIMP) (1984)



Figura: Apple  
Macintosh (1984)

- Ventanas, iconos, menús, apuntador (*Windows, Icons, Menus, Pointer*)
- 1984: Apple Macintosh, primer sistema WIMP con éxito comercial
- Multiprocesos, *no multitarea*



## Multitarea preventiva (1985)

- Multitarea preventiva real: 1985 (Amiga, Atari ST)
  - Sin hardware de protección de memoria
- Los programadores *tienen que considerar la concurrencia*



Figura: Commodore Amiga 500 (1987)



## Multitarea preventiva (1985)

- Multitarea preventiva real: 1985 (Amiga, Atari ST)
  - Sin hardware de protección de memoria
- Los programadores *tienen que considerar la concurrencia*



Figura: Commodore Amiga 500 (1987)

```
Software Failure. Press left mouse button to continue.  
Guru Meditation #00000004.0000AAC0
```



Figura: La meditación del gurú

# La profesionalización del escritorio

- Fines de los 1980 — Intel 80486, Motorola 68040, PowerPC:  
Hardware tan capaz como el de las *estaciones de trabajo*
- Reducción de las arquitecturas alternativas
- Reemplazo paulatino de los sistemas operativos por otros más capaces (o mejor *mercadeados*)
  - Mención breve de casos: DOS, Windows y OS/2; AmigaOS y Atari ST; NeXT y MacOS



## Convergen Unix y las computadoras *humildes*

- Unixes históricos para computadoras personales: Xenix, A/UX, SCO.
  - Muy limitados por su hardware
  - Precio desproporcionadamente alto; mejor ir por una *estación de trabajo*
- Génesis del software libre *ideológico*: GNU (1984)
- 386/BSD: Primer sistema *casi* libre
- Linux (1991+); GNU/Linux
- El mundo \*BSD
- ... Eventual muerte de las estaciones de trabajo
  - Apollo, Digital, Sun, SGI, HP (PA/RISC), ...



# Índice

- 1 ¿Qué son y qué hacen?
- 2 Historia y evolución
- 3 Computadoras personales
- 4 Dispositivos móviles



## El mercado de los dispositivos (*appliances*)

- 1990s: Agendas digitales inteligentes
  - Que se van *inflando* en computadoras completas
- 2000s: Ruteadores, modems inteligentes, controladores de TV (*set-top boxes*)
- 2007+: Teléfonos celulares *inteligentes*, tabletas — Por fin exitosos (tras incontables fracasos)

¿Cuál es la diferencia entre estos equipos y nuestras computadoras de escritorio?



## ¿Qué define a los *dispositivos móviles*?

- Bajo consumo eléctrico
  - Amplios *estados de ahorro de energía*
- Interfaz usuario limitada
  - Mecanismos de entrada reducidos
  - Adecuación a dicha realidad en la interfaz usuario
- Equipos ~~limitados en rendimiento~~



## ¿Qué define a los *dispositivos móviles*?

- Bajo consumo eléctrico
  - Amplios *estados de ahorro de energía*
- Interfaz usuario limitada
  - Mecanismos de entrada reducidos
  - Adecuación a dicha realidad en la interfaz usuario
- Equipos ~~limitados en rendimiento~~ ¿En serio?



## Primeros modelos

Primer computadora *portátil*: IBM 5100 (1975, 25Kg, pantalla de 5 pulgadas, US\$9,000)



IBM 5100 (CC-BY Sandstein)



1984: *Psion Organiser*: Computadora de bolsillo con reloj, calculadora, base de datos, cartuchos de aplicaciones. 4KB RAM, 2KB ROM, sin sistema operativo, programada en ensamblador. Antecedente directo de *Symbian*.



## 1990s: El *Asistente Digital Personal* (PDA)



Sharp Wizard



Apple Newton

1990s: Popularización de los *Asistentes Digitales Personales* (PDAs) — Agendas de teléfonos, calendario, calculadora... (No programables, rara vez expandibles)

Comenzó a desarrollarse la tecnología de entrada por pantalla, con interfaces táctiles con *stylus*, con grados de éxito... Variables.

Aparece el soporte para *ecosistemas* de software, comunicación entre dispositivos, modos de *hibernación*.



Palm Pilot



## 1995 – 2010: Puliendo la telefonía



Kyocera 6035



Nokia N810

- Aparición de los *teléfonos inteligentes*
- Personalidad dividida, dos computadoras en consonancia (señalización celular e interfaz usuario)
- Tendencia hacia interfaces *multitouch*
- Enfoque en eficiencia de consumo eléctrico



Palm Treo



iPhone

## 2010– Maduración del mercado móvil

- Consolidación de sistemas operativos disponibles
  - Dominantes: Android (← Linux), iOS (← MacOS)
  - Desaparecen: Symbian, Windows Phone, Firefox OS
- Universalización de la interfaz
  - ... Levante la mano quien *no* tiene uno consigo ...



## Características generales del segmento

- Almacenamiento en estado sólido (vs. discos magnéticos giratorios)
- Interfaz usuario: Multitarea, pero *monocontexto*
- Consumo eléctrico
  - Menor consumo
  - Más *estados de descanso*, detección de patrones de uso acorde
  - Salto constante entre estados
- No asumen estabilidad: Adecuación a un entorno cambiante
- Disponibilidad de aplicaciones: *Jardín amurallado*



# Arquitecturas hardware en boga hoy

- En escritorios y servidores: Derivada de PC (Intel x86)
- En dispositivos embebidos ARM (ocasionalmente MIPS; ojo OpenRISC)
  - ARM como alternativa de bajo consumo para servidores
- Sigue habiendo un importante espacio a controladores que no requieren sistema operativo (p.ej. Arduino, ARM perfiles R/M, ASICs. . . )
  - Probablemente siempre lo habrá

