Programación de Sistemas

Mtro. en IA José Rafael Rojano Cáceres
tareasrojano@gmail.com
http://www.uv.mx/rrojano

Taxonomía de los sistemas operativos

Referencia [Oney 96]

- La programación de sistemas tiene diferentes implicaciones relacionada con las características propias de los diferentes sistemas operativos.
- En cada uno de ellos se ofrecen diferentes funcionalidades que el desarrollador debe tomar en cuenta.
- A continuación se describirán algunas características con respecto a los S.O. de Windows de acuerdo con [Oney 96]

Características en Win3.1 (1/3)

- Diseñado para los 386
- SO de 16 bits, ejecutaba aplicaciones de 16 bits y aplicaciones DOS y aplicaciones del DOS extendidas
- API de 16 bits como interfaz con el sistema
- Interrupciones como interfaz con el DOS y el BIOS

La programación de sistemas en Win 3.1 *significó* trabajar con drivers para MS-DOS, utilerías TSR (Terminate and Stay Resident) y la DPMI (el DOS Protected Mode Interface)

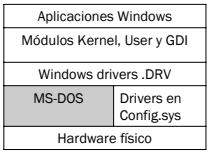
Características en Win3.1 (2/3)

- *App extendidas*. Aplicaciones de 16 o 32 bits construidas en el tope del DOS extender
  - Pueden correr en modo protegido mientras se encontraban en el DOS o llamaban a servicios del BIOS.
    - Kernel, User y GDI son aplicaciones de este tipo.
- *Modo real*. Modo operativo para los microprocesadores 80x86.
  - Provee un ambiente monotarea para las aplicaciones con acceso libre a la memoria y a los dispositivos de entrada y salida.

Características en Win3.1 (3/3)

- *Modo protegido*. Modo operativo nativo para los microprocesadores 80286 y superiores.
  - Soporta multitarea, seguridad en los datos y memoria virtual. (otra lectura [Englander 02, Pág. 407])

Arquitectura de Win3.1



Problemáticas

- Las aplicaciones pueden hacer que el SO se cuelgue
- El diseño de los módulos USER y GDI propios del SO poseen un heap limitado acarreando problemas de ejecución para las aplicaciones

Características en Win95 (1/3)

- Win 95 fue basado en la misma tecnología de máquina virtual que el Win 3.1. También permite la fácil adición de nuevo hardware.
- Una característica importante es la existencia del PSP (Program Segment Prefix).
  - Área de datos que MS-DOS usa para distinguir entre aplicaciones.
  - Más importante el PSP permite manejar archivos
  - Incluso las aplicaciones de 32 bits tienen PSP en Win 95, éste se encuentra en el primer megabyte de la memoria.

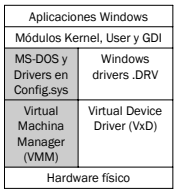
Características en Win95 (2/3)

- SO de 16 bits capaz de correr aplicaciones de 16 y 32 bits.
- Capaz de proveer hasta 4 GB de memoria virtual (dependiendo del espacio y la memoria física)
- Capaz de ejecutar multitasking.
- Se ejecuta en lo que se denomina “modo mejorado 386”.
- No provee un ambiente protegido a diferencia de NT donde datos y programas sean separados.

Características en Win95 (3/3)

- Muchos de los componentes de SO corren al nivel del anillo cero y usan el esquema de memoria de plana (flat).
- Las aplicaciones corren al nivel del anillo tres.
- Los módulos Kernel, USER y GDI son los descriptores del SO, sin embargo son aplicaciones que corren al nivel de anillo tres.
- El verdadero sistema operativo de Win 95 y Win 3.x es provisto por el Virtual Machine Manager (VMM).

Arquitectura de Win95



Windows NT

- Verdadero SO de 32 Bits
- Creado a partir de cero
- orientado a la **multiprogramación** (varios usuarios trabajando en la misma máquina al mismo tiempo)
- Capacidades **multiproceso** en donde cada proceso cuenta con su propio espacio de direcciones virtual paginado por demanda. [Tanenbaum 00, Pág. 449-455]

Arquitectura de WinNT (1/4)

- Su arquitectura fue diseñada a partir de un núcleo o kernel más o menos pequeño.
- Su arquitectura se divide en capas o módulos:
  - usuario, kernel y ejecutivo
- En la parte inferior de la arquitectura se encuentra la capa HAL o capa de abstracción del hardware.
  - Su función es presentar al SO una serie de dispositivos sin sus particularidades, con ello puede existir una portabilidad del SO hacia cualquier plataforma

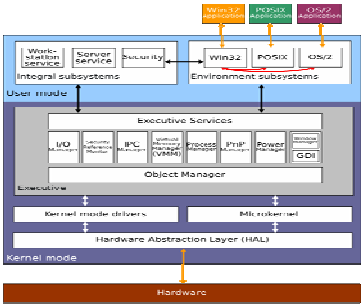
Arquitectura de WinNT (2/4)

- La **capa de kernel** contiene el micronucleo y los drivers de dispositivos.
- El micronucleo y los drivers tienen acceso directo al hardware.
- El micronucleo maneja las interrupciones, trampas y excepciones, la planificación y la sincronización.
- El micronucleo siempre se encuentra alojado en memoria y no se puede eliminar, aunque puede ceder el control temporalmente a las interrupciones.

Arquitectura de WinNT (3/4)

- La **capa del ejecutivo** es independiente de la arquitectura y se puede trasladar a otras arquitecturas.
  - El ejecutivo se compone de tres sub capas: capa de sistemas de archivos, capa de gestor de objetos y la tercera que a su vez se compone de 6 elementos.
- Fuera de las capas anteriores se encuentra la **capa de usuario** en donde residen los programas de usuario y los subsistemas de entorno.

Arquitectura de WinNT (4/4)



Problemática

- Un aspecto interesante es conocer que servicios ofrece cada una de estas capas a detalle, esto permitiría ser el mecanismo de interacción directa con el programador, pero dado que Microsoft nunca a publicado la lista completa de los mismos y además varían de versión a versión se hace imposible realizar este proceso.
- En vez de ello se cuenta con un **API Win32** (*Application Program Interface*). El API es un conjunto de procedimientos de biblioteca empleados para realizar llamadas al sistema.

Tareas

- Leer y reportar la estructura del PSP
- Características, modo de acceso del DMPI
  - Entregar (Lunes 17)

Los sabores de Windows y los controladores

Introducción

- A la fecha de la aparición de Win95 el desarrollo de la programación de sistemas se oriento a escribir código de 32 bits para los módulos del anillo cero conocidos como Vxd (Virtual Device Driver).
- En un principio la tarea de los Vxd era *la virtualización* del hardware para los módulos del anillo tres tales como drivers para MS-Dos y Dll de Windows. En Win95 los Vxd son la interfaz primaria con el hardware, también sirven para distintas aplicaciones necesarias para los servicios.

Anillos del SO

- *Software de anillo cero*. Software que se ejecuta con el privilegio más alto en el microprocesador Intel.
- *Software de anillo tres*. Software que se ejecuta con el privilegio más bajo en el microprocesador Intel.
  - Para referencia de los privilegios véase [Tischer 96, Pág. 448] [Tanenbaum 00, Pág. 425]

Modo real de Windows

- El MS-DOS y el BIOS han sido los proveedores de los controladores para muchos dispositivos desde el principio.
- El Bios realiza esta tarea a través de unas pocas pero bien conocidas interrupciones:
  - Int 10. Subsistema de Video
  - Int 13. Subsistema para los discos.
  - Int 16. Subsistema para el teclado.
- El Bios también se encarga de lidiar con las interrupciones de hardware y asume la responsabilidad para manejar el Programmable Interrupt Controler (PIC)
- El MS-DOS también exporta un conjunto de bien conocidas interrupciones como son:
  - Int 21, Int 25, Int 26.
  - Entre muchas otras.
- Provee también un mecanismo para cargar controladores al momento del boot (argumento *device=* del *config.sys*).

Modo estándar de Windows (1/2)

- El modo de operación conocido como modo estándar de Windows consistía en la posibilidad de que Windows cambiará a operar en modo protegido, pero manteniendo el uso de MS-DOS y el Bios para todas sus necesidades de sistema.
- Windows proveyó un mecanismo para lograr rápidamente este cambio, sin embargo no el modo inverso, con lo cual la única forma era resetear el procesador para regresar al modo real (ctrl+alt+del).

Modo estándar de Windows (1/2)

- Ya que para controlar dispositivos como teclado o ratón no se podría considerar en tener que estar cambiando de modo para leer los dispositivos se crearon los controladores de modo protegido que atendieran interrupciones.
- Tales dispositivos se almacenaron en el SO bajo la extensión .DRV. A estos drivers
  - Los DRV son aplicaciones del anillo tres
  - Lucen como las DLL de 16 bits.
  - El propósito de los mismos es ser un intermediario entre los módulos de User, GDI y Kernel sin abandonar el modo protegido.

Modo mejorado de Windows (1/2)

- Con la introducción del 80386 se pudo tener un tercer modo de operación, llamado modo mejorado, en el cual Windows usa paginación y una características llamada V86 para crear máquinas virtuales.
- Para una aplicación una VM luce como una máquina normal (CPU con teclado, ratón, monitor, etc).
- Diferentes VM pueden compartir la misma máquina física a través de un proceso denominado *virtualización*. Desde el punto de vista del usuario la mayor ventaja era la posibilidad de ejecutar sesiones de MS-DOS en el mismo escritorio.
- El encargado de soportar la virtualización son los Vxd

Modo mejorado de Windows (2/2)

- su nombre proviene de Virtual x device, queriendo decir que el driver virtualiza algún dispositivo representado por la x. por ejemplo un driver llamado vKd, virtualiza al teclado.
- Existen otra clase de controladores que no virtualizan dispositivos pero son una forma convenientemente repositorios de varios servicios de bajo nivel. Ejemplo de ello son:
  - Pageswap y Pagefile, en conjunto permiten administrar la memoria swap que se emplea en esta modalidad de Windows para aumentar la RAM.

Tareas

- Exponer acerca del API
  - Win16 (Martes 18)
  - Win32 [Conger 92]
- Exponer acerca del modelo de driver de Windows (links)
  - VxD (Viernes 21)
  - WDM (Lunes 24)
  - WDF (Martes 25)

Bibliografía

1. [Donovan 72] John Donovan, Systems Programming, McGraw Hill, 1972
2. [Englander 02] Irv Englander, Arquitectura computacional 2da Edición, CECSA, 2002
3. [Jurgens 91] David Jurgens, Help PC 2.10 software de referencia, 1991.
4. [Oney 96] Michael Oney, Systems Programming for Windows 95, Microsoft Press, 1996
5. [Powell 01] Robert Powell, C# and the .NET Framework The C++ perspective, Sams, 2001
6. [Tanenbaum 00] Andrew Tanenbaum, Organización de computadoras un enfoque estructurado, Pearson Education, 2000
7. [Tischer 96] Michael Tischer, PC Interno 5, Marcombo, 1996.
8. [Duran 07] Luis Duran Rodriguez, El gran libro del PC Interno, AlfaOmega, 2007
9. [Conger 92] Conger, James L. , Windows API bible : the definitive programmer's reference, Waite Group Pr

Ligas útiles

- Preguntas frecuentes sobre el API Win32
  - http://web.archive.org/web/20051230123613/www.iseran.com/Win32/FAQ/faq.htm
- Drivers en Windows
  - (WDM) http://www.microsoft.com/whdc/archive/wdmoverview.mspx
  - (WDF) http://www.microsoft.com/whdc/driver/wdf/default.mspx
  - http://www.microsoft.com/whdc/driver/kernel/KB-drv.mspx
  - http://en.wikipedia.org/wiki/Windows_Driver_Foundation