WINDOWS POR DENTRO

AUTOR: MAX JOSÉ BERMÚDEZ LEÓN

DICIEMBRE: 2020

San Marcos

Introducción

Uno de las características que Windows comparte con el resto de los Sistemas Operativos avanzados es la división de tareas del Sistema Operativo en múltiples categorías, las cuales están asociadas a los modos actuales soportados por los microprocesadores. Estos modos proporcionan a los programas que corren dentro de ellos diferentes niveles de privilegios para acceder al hardware o a otros programas que están corriendo en el sistema. Windows usa un modo privilegiado (Kernel) y un modo no privilegiado (Usuario).

Uno de los objetivos fundamentales del diseño fue el tener un núcleo tan pequeño como fuera posible, en el que estuvieran integrados módulos que dieran respuesta a aquellas llamadas al sistema que necesariamente se tuvieran que ejecutar en modo privilegiado (modo kernel). El resto de las llamadas se expulsarían del núcleo hacia otras entidades que se ejecutarían en modo no privilegiado (modo usuario), y de esta manera el núcleo resultaría una base compacta, robusta y estable.

Contenido

Arquitectura de Windows

Con el paso de los años se ha producido una evolución gradual de la estructura y capacidades de los Sistemas Operativos. Sin embargo, recientemente se ha introducido un cierto número de nuevos elementos de diseño en los nuevos Sistemas Operativos y en las nuevas versiones de los Sistemas Operativos existentes.

Uno de los objetivos fundamentales del diseño fue el tener un núcleo tan pequeño como fuera posible, en el que estuvieran integrados módulos que dieran respuesta a aquellas llamadas al sistema que necesariamente se tuvieran que ejecutar en modo privilegiado (modo kernel). El resto de las llamadas se expulsarían del núcleo hacia otras entidades que se ejecutarían en modo no privilegiado (modo usuario), y de esta manera el núcleo resultaría una base compacta, robusta y estable.

Los Sistemas Operativos modernos responden a nuevos desarrollos del hardware y nuevas aplicaciones. Entre estos dispositivos de hardware están las máquinas multiprocesador, incrementos enormes de la velocidad de la máquina, alta velocidad en los enlaces de las redes de comunicación e incremento en el tamaño y variedad de los dispositivos de almacenamiento de memoria.

En los campos de aplicación que han influido en el diseño de los Sistema Operativos están las aplicaciones multimedia, el acceso a Internet y páginas Web y la ejecución cliente/servidor.

El porcentaje de cambios en las demandas de los Sistemas Operativos, requiere no solamente las modificaciones y mejoras en las arquitecturas ya existentes, sino nuevas formas de organización del Sistema Operativo. Muchos de los diferentes enfoques y elementos de diseño se han probado tanto en Sistemas Operativos experimentales como comerciales, y muchos de ellos encajan dentro de las siguientes categorías:

- Arquitectura Micronúcleo.
- Multihilos.
- Multiproceso Simétrico.
- Sistemas Operativos Distribuidos.
- Diseño Orientado a Objeto.

La mayor parte de los Sistemas Operativos hasta hace poco tiempo se caracterizaban por un gran núcleo monolítico. Gran parte de la funcionalidad que se pensaba debía tener un Sistema Operativo la proporcionaba este gran núcleo, incluyendo planificación, sistema de archivos, redes, controladores de dispositivos, gestión de memoria y muchas cosas más. Normalmente un núcleo monolítico está implementado como un único proceso, con todos sus componentes compartiendo el mismo espacio de direcciones.

La arquitectura micronúcleo asigna solamente unas pocas funciones esenciales al núcleo, incluyendo espacios de direcciones, comunicación entre procesos (IPC) y planificación básica. Otros servicios del Sistema Operativo los proporciona procesos, algunas veces llamados servidores, que se ejecutan en modo usuario y que el micronúcleo trata como a cualquier otra aplicación. Este enfoque desconecta el núcleo y el desarrollo de servidores.

Los servidores pueden estar diseñados para aplicaciones específicas o necesidades del entorno. El enfoque del micronúcleo simplifica la implementación, proporciona flexibilidad y se adapta bien para entornos distribuidos. En esencia, un micronúcleo interactúa de la misma forma con procesos servidores locales y remotos, facilitando la construcción de sistemas distribuidos.

Uno de los pasos más importantes que revolucionó los Sistemas Operativos de la Microsoft fue el diseño y creación de un Sistema Operativo extensible, portable, fiable, adaptable, robusto, seguro y compatible con sus versiones anteriores (Windows NT).
Y para ello crearon la siguiente arquitectura modular:

La cual está compuesta por una serie de componentes separados donde cada cual es responsable de sus funciones y brindan servicios a otros componentes. Esta arquitectura es del tipo cliente – servidor ya que los programas de aplicación son contemplados por el sistema operativo como si fueran clientes a los que hay que servir, y para lo cual viene equipado con distintas entidades servidoras.

Ya creado este diseño las demás versiones que le sucedieron a Windows NT fueron tomando esta arquitectura como base y le fueron adicionando nuevos componentes.
Uno de las características que Windows comparte con el resto de los Sistemas Operativos avanzados es la división de tareas del Sistema Operativo en múltiples categorías, las cuales están asociadas a los modos actuales soportados por los microprocesadores. Estos modos proporcionan a los programas que corren dentro de ellos diferentes niveles de privilegios para acceder al hardware o a otros programas que están corriendo en el sistema. Windows usa un modo privilegiado (Kernel) y un modo no privilegiado (Usuario).

Uno de los objetivos fundamentales del diseño fue el tener un núcleo tan pequeño como fuera posible, en el que estuvieran integrados módulos que dieran respuesta a aquellas llamadas al sistema que necesariamente se tuvieran que ejecutar en modo privilegiado (modo kernel). El resto de las llamadas se expulsarían del núcleo hacia otras entidades que se ejecutarían en modo no privilegiado (modo usuario), y de esta manera el núcleo resultaría una base compacta, robusta y estable.

El Modo Usuario es un modo menos privilegiado de funcionamiento, sin el acceso directo al hardware. El código que corre en este modo sólo actúa en su propio espacio de dirección. Este usa las APIs (System Application Program Interfaces) para pedir los servicios del sistema.

El Modo Kernel es un modo muy privilegiado de funcionamiento, donde el código tiene el acceso directo a todo el hardware y toda la memoria, incluso a los espacios de dirección de todos los procesos del modo usuario. La parte de WINDOWS que corre en el modo Kernel se llama Ejecutor de Windows, que no es más que un conjunto de servicios disponibles a todos los componentes del Sistema Operativo, donde cada grupo de servicios es manipulado por componentes que son totalmente independientes (entre ellos el Núcleo) entre sí y se comunican a través de interfaces bien definidas.

Todos los programas que no corren en Modo Kernel corren en Modo Usuario. La mayoría del código del Sistema Operativo corre en Modo Usuario, así como los subsistemas de ambiente (Win32 y POSIX) y aplicaciones de usuario. Estos programas solamente acceden a su propio espacio de direcciones e interactúan con el resto del sistema a través de mensajes Cliente/Servidor.

Capa de Abstracción de Hardware (HAL).

Conocido por sus siglas en inglés HAL (Hardware Abstraction Layer) es una interfaz entre el hardware y el resto del Sistema Operativo, está implementada como una biblioteca de enlace dinámico (dll) y es responsable de proteger el resto del sistema de las especificaciones del hardware, tales como controladores de interrupción e interfaces de entrada/salida. Esta abstracción hace al sistema más portable ya que el resto del sistema no tiene que preocuparse sobre que plataforma está corriendo. Cada plataforma en que el sistema corre necesita un HAL específico. El diseño intenta que cuando Windows sea portado a una nueva arquitectura de procesador, el HAL sea reescrito para el nuevo procesador, pero el resto del sistema simplemente debe ser recompilado.

Este también suministra la interfaz para el multiprocesamiento simétrico (conocido por sus siglas en inglés SMP). Las versiones Server contienen dos HALs para arquitectura de procesador (Intel, MIPS, PowerPC y and Alpha), el primero es usado para soportar un solo procesador, mientras que el segundo soporta hasta cuatro procesadores.

Para cada procesador físico que existe en la computadora el HAL representa un procesador virtualizado al microkernel. La idea es que el procesador virtualizado esconda las características especiales del propio procesador al sistema operativo, quiere esto decir que si por ejemplo se tiene dos sistemas multiprocesadores, uno corriendo sobre un procesador Intel y otro corriendo con un Alpha, los HALs en cada sistema serían diferentes, pero los procesadores virtualizados que este presenta al microkernel en ambos casos pudieran ser

idénticos. Sobre un sistema SMP (Multiprocesamiento Simétrico) para cada procesador físico en el sistema el HAL representa un procesador virtualizado al microkernel.

A este componente solo pueden acceder componentes del Ejecutor de Windows y nunca se llama por los programas del Modo Usuario. El HAL también intenta ser la única pieza de software dentro del sistema que se comunique con el hardware, la ventaja de esto es que otros programas no pueden escribir información en el hardware ni accidentalmente, ni intencionalmente y causar una caída del sistema, también impidiendo que programas lean información directamente del hardware.

MicroKernel

Es el responsable de todas las acciones que se realizan sobre le sistema y casi todas las funciones del sistema pasan a través de él.
El diseño de este componente asigna muchas de las funciones normalmente asignadas al Kernel en los Sistemas Operativos tradicionales a un grupo de programas llamado Ejecutor de Windows, del cual el microkernel es parte, corre en el modo privilegiado y ambos (el ejecutor y el microkernel) se comunican a través de primitivas del sistema operativo a bajo nivel.

La principal tarea de este componente es la planificación de ejecución de hilos (segmento de código perteneciente a un proceso particular). A cada hilo es asignada una prioridad de 0 a 31, este entonces envía hilos a correr en dependencia de su número de prioridad y los permite ejecutarse un tiempo determinado antes de apropiarse de ellos y permitir que otro proceso corra.

El Ejecutor de Windows.
El Ejecutor de Windows se encarga de las tareas importantes, las que son de vital importancia para el sistema completo, ya que el microkernel está casi siempre demasiado

ocupado para dirigirse directamente.
Una definición clara es que el Ejecutor de Windows provee los fundamentos del sistema operativo que serán suministradas a todas las aplicaciones que corren sobre el sistema. Este incluye servicios como la Administración de Objetos, de Memoria virtual, de Entrada-Salida y de Procesos.

El Ejecutor de Windows corre exclusivamente en Modo Kernel y es llamado por los subsistemas de ambiente protegido cuando estos necesitan de sus servicios. Debido a la jerarquía de Windows las aplicaciones que corren en Modo Usuario no pueden llamar segmentos del Ejecutor de Windows directamente, sino servicios de demanda de los subsistemas de ambiente (explicado en capítulos posteriores), como Win32 y POSIX los que a su vez se encargan de llamar los componentes del Ejecutor de Windows.
El Administrador de Objetos (Object Manager) es usado para crear, modificar y eliminar objetos (tipos de datos abstractos que son usados para representar recursos del Sistema Operativo) usados por todos los sistemas que conforman el Ejecutor de Windows. Este también proporciona información sobre el estado de los objetos a todo el Sistema Operativo. Los objetos pueden ser cosas concretas, tales como puertos de dispositivos, o pueden ser más abstractos como hilos. Cuando un objeto es creado a este se le da un nombre por el cual otros programas pueden accederle. Cuando un proceso necesita acceder al objeto este solicita un tratamiento de objeto al administrador de objetos.

El Administrador de Procesos.

El Administrador de Procesos (Process Manager) es el responsable de crear, quitar y modificar los estados de todos los procesos e hilos. Este también proporciona información sobre el estado de procesos e hilos al resto del sistema.

Un proceso, por la definición, incluye un espacio de dirección virtual, uno o más hilos, un segmento de código del programa ejecutable, y un conjunto de recursos del sistema. Un hilo es un objeto ejecutable que pertenece a un solo proceso y contiene a un contador del

programa que apunta a su posición actual en el segmento de código ejecutable del proceso, dos pilas, y un conjunto de valores del registro.

El Administrador de Procesos, como todos los miembros del Ejecutor de Windows, juega un papel vital en el funcionamiento del sistema entero. Cuando una aplicación comienza su ejecución, se crea como un proceso lo que requiere una llamada al Administrador de Procesos. Como todo proceso debe tener por lo menos un hilo, el Administrador de Procesos es invocado de nuevo para crear el hilo.

El Administrador de Memoria Virtual.

El Administrador de Memoria Virtual (Virtual Memory Manager o VMM) proporciona la gestión de memoria virtual del sistema. La memoria virtual es un esquema que permite usar los recursos del disco en lugar de la memoria física del sistema moviendo las páginas al disco cuando estas no están siendo usadas y recuperándolas cuando se les necesitan. Este es un segmento integral de Windows el cual asigna espacios de direcciones de 32 bit a cada proceso sin preocuparse de la cantidad de memoria física del sistema.

Servicios de Llamadas a Procedimientos Locales.

El Servicio de Llamadas a Procedimientos Locales (Local Procedure Call Facility o LPC) se integran al diseño cliente/servidor de Windows. Este es la interfaz entre todos los procesos clientes y servidores que corren localmente en el sistema.

La estructura del Servicio de Llamadas a Procedimientos Locales es muy similar a la de las llamadas a Procedimientos Remotos (RPC), excepto que esta está optimizada y solamente soporta comunicación entre procesos clientes y servidores localmente. Más específicamente, el LPC es un mecanismo que permite a dos hilos en procesos diferentes intercambiar información.

El Monitor de Seguridad.

El Monitor de Seguridad (Security Reference Monitor o SRM) es el lecho de toda la seguridad dentro del sistema WINDOWS y es el responsable de hacer cumplir todas las políticas de seguridad en la computadora local.

Este componente trabaja conjuntamente con los subsistemas de tiempo de corrida, proceso de conexión al sistema (conocido como logon process) y control de la seguridad local (local security authority). Cuando un usuario intenta conectarse al sistema su identidad es verificada, el subsistema de proceso de conexión pide una ficha de acceso de seguridad (conocido por sus siglas en inglés SAT o security access token) del usuario. El SAT contiene una lista de los privilegios de usuarios y grupos. Este se usa como llave para ese usuario durante la sesión de conexión. Siempre que el usuario quiera hacer algo, el SAT es presentado y usado para determinar si el usuario puede realizar las acciones.

El Administrador de Entrada-Salida.

El Administrador de Entrada-Salida (I/O Manager) es responsable de gestionar la comunicación entre los distintos drivers de dispositivo, para lo cual implementa una interfaz bien definida que permite el tratamiento de todos los drivers de una manera homogénea, sin que intervenga el cómo funciona específicamente cada uno. Tiene una serie de subcomponentes que son:
Driver del Sistema de Archivos: este se encarga de establecer la comunicación con los drivers de los Sistemas de Ficheros, ya que el sistema permite la coexistencia de múltiples Sistemas de Archivos en diferentes particiones lógicas de la misma unidad física.
El servidor y el redirector de red.
Los drivers de dispositivo del sistema.

El administrador de caches (Cache Manager): este se encarga de manipular la cache para todo el Sistema de Entrada y Salida. Este es un método que utilizan los sistemas de archivos para mejorar su rendimiento, donde en lugar de leer y escribir en disco un fichero usado

frecuentemente este se almacena en una cache de memoria y la lectura y escritura de estos ficheros se realiza desde memoria. Este componente se encarga de la magia negra que es a menudo necesaria para hacer que varios dispositivos se comuniquen entre sí y convivan juntos en un segmento. El Administrador de Entrada-Salida (I/O Manager) es responsable de gestionar la comunicación entre los distintos drivers de dispositivo.
Subsistemas de Ambiente Protegido

Dos de los objetivos de WINDOWS son personalidad y compatibilidad. Esto ha sido logrado a través de los subsistemas de ambiente protegido.

La personalidad esencialmente significa que WINDOWS expone múltiples conjuntos de interfaces de programas de aplicación (APIs) y puede actuar eficazmente como si fuera un sistema operativo diferente. WINDOWS viene con una personalidad POSIX y OS/2 además de sus personalidades Win32, Win16 y DOS.

En WINDOWS, hay tres subsistemas de ambiente protegido:

- El subsistema de Win32
- El subsistema de POSIX
- El subsistema de OS/2

Aunque algunas veces se muestran las personalidades Win16 y DOS incluidas en una lista de subsistemas de ambiente protegido, ellas realmente son parte del subsistema Win32.
Los subsistemas de ambiente protegido actúan como los mediadores entre las aplicaciones del Modo Usuario y el Ejecutor de Windows.
Recuerde que el Ejecutor de Windows y todos sus componentes viven en el Modo Privilegiado o Modo Kernel, mientras que todos los demás viven en el Modo Usuario, esto incluye todos los subsistemas de ambiente. Cuando una aplicación hace una llamada a un subsistema de ambiente, este es pasado a través de una capa de servicios del Ejecutor de Windows.
Cada subsistema de ambiente guarda huella de sus propios procesos y trabaja

independientemente de los otros subsistemas. Cada aplicación sólo puede correr en el subsistema para el cual fue diseñado. Cuando usted inicia una aplicación en WINDOWS, mira el encabezamiento representado por el archivo y determina en cuál subsistema ejecutar la aplicación.

El Subsistema Win32

Win32 es el subsistema nativo y primario de WINDOWS. Las bases para este subsistema es el conjunto de APIs de Win32. Muchos de estas API son extensiones directas de sus homólogas Win16.
Este subsistema actúa como un servidor para todos los otros subsistemas de ambiente soportados en WINDOWS, los que actúan como clientes y traducen sus llamadas API hacia las API apropiadas de Win32.

El subsistema Win32 es responsable de toda la entrada y salida. Este posee el control de la pantalla, el teclado, y el ratón. Cuando otros subsistemas, como OS/2 o POSIX, necesitan beneficiarse de estos dispositivos, ellos piden los servicios al subsistema de Win32.
Algunos de los objetivos que se trazaron para mantener la compatibilidad con las aplicaciones hechas en versiones anteriores fueron:
Permitir que los programas hechos sobre DOS pudieran correr sin modificación.
Suministrar la capacidad para ejecutar la mayoría de las aplicaciones Windows de 16 bits sin modificación
Proteger al sistema y otras aplicaciones de 32 bits de la interferencia de las aplicaciones de 16 bits y DOS.

Permitir a las plataformas RISC (Reduced Instruction set Computer, microprocesador cuyo número de instrucciones es reducido para lograr una frecuencia más alta de trabajo) ejecutar aplicaciones Windows de 16 bits y DOS.
Suministrar un mecanismo para compartir datos entre aplicaciones Windows de 32 y 16 bits.

El Subsistema POSIX.

Microsoft prestó mucha atención a los diferentes estándares de sistemas abiertos cuando Windows NT estaba en vía de desarrollo. Ellos reconocieron el valor de soportar sistemas abiertos como un método para ganar aceptación de su nuevo sistema operativo avanzado dentro del mercado.

Uno de los estándares más frecuentemente citados soportados por Windows es el POSIX (Interfaz de Sistema operativo Portable Basado en Unix), el cual representa la interfaz del Sistema Operativo portable y fue desarrollado por el IEEE (Instituto de Ingenieros en Electricidad y Electrónica) como un método de proporcionar portabilidad a las aplicaciones hechas sobre plataformas UNIX. No obstante, POSIX se ha integrado en muchos sistemas no UNIX.

El Subsistema OS/2.

El subsistema de OS/2 está implementado como un subsistema de ambiente protegido, parecido al subsistema POSIX. Este traduce las llamadas API de OS/2 en llamadas a APIs de Win32 que son servidas por el subsistema de Win32.

El subsistema y sus aplicaciones corren en su propio espacio de memoria protegido de 32 bits y constituyen multitarea preventiva unas respecto a otras y respecto a otras aplicaciones que corren en el sistema.

Además de un conjunto de motores APIs de OS/2, el subsistema implementa muchos APIs gestores de LAN (Red de Área Local), incluyendo tuberías, NETBIOS y mailslots. De esta manera difiere del subsistema POSIX ya que este no posee soporte para gestión de redes.

El Subsistema OS/2 igual que el subsistema POSIX proporciona un entorno para aplicaciones UNIX, este subsistema da soporte a las aplicaciones OS/2. Proporciona la interfaz gráfica y las llamadas al sistema; las llamadas son servidas con ayuda del Ejecutor de Windows.

Windows es un sistema que aprovecha la potencia de los procesadores, ha sido diseñado para adaptarse a las nuevas tecnologías, ofrece compatibilidad con varias plataformas

(OS/2, Unix y versiones anteriores a el mismo), soporta el multiprocesamiento simétrico, buen rendimiento y conectividad, seguridad y al no estar encasillado en ningún modelo estandar de Sistema Operativo tiene la capacidad de combinar las ventajas del modelo cliente/servidor, puede correr además sobre múltiples arquitecturas con un mínimo de cambios, permite que varios procesos sean ejecutados simultáneamente en varios procesadores y estos no se apropien de recursos del sistema por tiempo indefinido, sino por tratamiento del sistema.

Sistemas de archivos

Como vimos en el módulo dos cuando se estaba estudiando Linux, se dio una reseña de lo que es un sistema de archivos, pero vamos refrescarlo, entonces podemos decir que Un sistema de archivos es el sistema de almacenamiento de un dispositivo de memoria, que estructura y organiza la escritura, búsqueda, lectura, almacenamiento, edición y eliminación de archivos de una manera concreta. El objetivo principal de esta organización es que el usuario pueda identificar los archivos sin lugar a error y acceder a ellos lo más rápido posible. Los sistemas de archivos también otorgan a los archivos, entre otras, las siguientes características:

- Convenciones para nombrar a los archivos
- Atributos de archivo
- Control(es) de acceso

Asimismo, los sistemas de archivos son un componente operativo importante, ya que actúan como una interfaz entre el sistema operativo y todos los dispositivos conectados al equipo (internos y externos, como las memorias USB, o los discos duros externos).

Los sistemas de organización de archivos que emplea Microsoft Windows utilizan el acceso secuencial indexado (acceso secuencial y acceso indexado adjuntos en un mismo método), el acceso directo en algunos casos en la utilización de los sistemas de organización por tablas.

- Tabla de asignación de archivos: comúnmente conocido como FAT (del inglés File Allocation Table), es un sistema de archivos. Es un formato popular para disquetes admitido prácticamente por todos los sistemas operativos existentes para computadora personal. Se utiliza como mecanismo de intercambio de datos entre sistemas operativos distintos que coexisten en la misma computadora, lo que se conoce como entorno multiarranque.
- HPFS (High Performance File System).
- NTFS (New Technology File System).

NTFS:

NTFS son las siglas de New Technology File System, es decir, sistema de archivos de nueva tecnología. Se trata de un sistema de archivos muy extendido gracias a la popularidad de Microsoft y que sirve para organizar datos en discos duros y otros soportes de almacenamiento. Desde el lanzamiento de Windows XP en el año 2001, NTFS es el estándar obligatorio en los sistemas operativos Windows. En este artículo te contamos cómo funciona, qué ventajas tiene y cómo se diferencia de otros sistemas como, por ejemplo, FAT.
El tamaño máximo de una partición en el sistema NTFS es de unos 2 terabytes. El sistema no presenta restricciones para archivos sueltos, de manera que, en teoría, sería posible guardar un solo archivo de casi 2 terabytes en un soporte de datos formateado con NTFS. El tamaño del clúster de NTFS ha sido ampliado notablemente en comparación con los sistemas de archivos clásicos como FAT32 y es de aproximadamente 16×10^18. En el sistema de archivos FAT32, en cambio, esta cifra se reduce a 4 294 967 296. Según el estándar NTFS, un nombre de archivo puede contener, como máximo, 255 caracteres.

NTFS sigue el principio de todo en un archivo. En cambio, otros sistemas de archivos, como los de los sistemas operativos Unix, por ejemplo, trabajan según el principio todo es un archivo. En NTFS, todos los datos relativos a los archivos guardados se registran en la tabla maestra de archivos o Master File Table (MFT). Se trata de un índice que contiene, entre otras cosas, información acerca de qué bloques del soporte de almacenamiento pertenecen a qué archivos y qué permisos de acceso y atributos tiene cada archivo. En el sistema de archivos NTFS, la tabla maestra guarda atributos como el tipo y el tamaño de archivo, la fecha de creación y la de la última modificación. La MFT dispone por tanto de una posición especial en los soportes de datos con formato NTFS: para ella se suele reservar el 12,5 % del tamaño de la partición. Este porcentaje del espacio no puede ser ocupado por otros archivos. La fragmentación del soporte de datos empieza en cuanto la MFT está totalmente llena de datos.

Desde Windows XP, NTFS ha sido el estándar de preferencia en los sistemas de Microsoft y, desde Windows Vista, el disco duro que contiene el sistema operativo ha de estar obligatoriamente formateado en NTFS, lo cual es comprensible: en comparación con las antiguas versiones de FAT, como FAT32 o FAT16, NTFS ofrece ventajas importantes.
Por lo general, el sistema de archivos NTFS funciona particularmente bien si se usa en redes, porque es aquí donde su estructura, bien organizada, incluyendo el práctico control de acceso a las funciones de lectura y de escritura por parte de los usuarios, cobra especial sentido. Si se compara con FAT32, el estándar que lo precede y que aún se usa hoy en día en ciertas situaciones, NTFS ofrece además otras ventajas: el tamaño máximo de las particiones es mucho mayor, de aproximadamente 16 terabytes. Se trata de una cifra que incluso actualmente (en 2020) apenas superan los discos duros disponibles en el mercado, ya sean los clásicos de tipo mecánico (HDD) o las modernas memorias flash de tipo SSD.

NTFS puede escribir archivos pequeños bastante más rápido que un sistema de archivos como FAT32. Además, no tiene limitaciones respecto al tamaño de los archivos. Gracias a una selección inteligente de los sectores que se ocupan, el sistema de archivos atenúa el problema de la fragmentación y reduce la necesidad de desfragmentar el disco constantemente. Con NTFS también se producen menos pérdidas de datos, ya que el sistema reconoce rápidamente los sectores dañados y retira los archivos que contienen.
Asimismo, gracias a NTFS, además de los nombres de archivo, también pueden registrarse otros tipos de información, con un tamaño de hasta 64 kibibytes (KiB).
Los metadatos guardados de esta manera muestran claramente con qué programa puede abrirse un archivo y cuentan con la ventaja adicional de que con NTFS no es necesario indicar la extensión del archivo. Con la información de todos los metadatos se registra un diario o journal: cuando se planea una acción, esta se registra primero en el diario, luego se edita el acceso de escritura y, finalmente, se actualiza el diario. De esta forma, pueden evitarse muchas incoherencias, ya que, incluso en caso de avería o apagón, basta con corregir el diario.

FAT

FAT es con diferencia el sistema de archivos más simple de aquellos compatibles con Windows NT. El sistema de archivos FAT se caracteriza por la tabla de asignación de archivos (FAT), que es realmente una tabla que reside en la parte más "superior" del volumen. Para proteger el volumen, se guardan dos copias de la FAT por si una resultara dañada. Además, las tablas FAT y el directorio raíz deben almacenarse en una ubicación fija para que los archivos de arranque del sistema se puedan ubicar correctamente.

Un disco con formato FAT se asigna en clústeres, cuyo tamaño viene determinado por el tamaño del volumen. Cuando se crea un archivo, se crea una entrada en el directorio y se establece el primer número de clúster que contiene datos. Esta entrada de la tabla FAT indica que este es el último clúster del archivo o bien señala al clúster siguiente.

La actualización de la tabla FAT es muy importante y requiere mucho tiempo. Si la tabla FAT no se actualiza con regularidad, podría producirse una pérdida de datos. Requiere mucho tiempo porque las cabezas lectoras de disco deben cambiar de posición y ponerse a cero en la pista lógica de la unidad cada vez que se actualiza la tabla FAT.

No hay ninguna organización en cuanto a la estructura de directorios de FAT, y se asigna a los archivos la primera ubicación libre de la unidad. Además, FAT solo es compatible con los atributos de los archivos de almacenamiento, del sistema, ocultos y de solo lectura.

Convención de nomenclatura de FAT

FAT usa la convención de nomenclatura tradicional 8.3 y todos los nombres de archivo deben crearse con el conjunto de caracteres ASCII. El nombre de un archivo o directorio puede tener ocho caracteres de longitud, después un separador de punto (.) y una extensión de hasta tres caracteres. El nombre debe empezar con una letra o un número y puede contener cualquier carácter excepto los siguientes:

. " / \ [ ] : ; | = ,

Si se usa cualquiera de estos caracteres, pueden producirse resultados inesperados. El nombre no puede contener espacios en blanco.

Los nombres siguientes están reservados:

CON, AUX, COM1, COM2, COM3, COM4, LPT1, LPT2, LPT3, PRN, NUL

Todos los caracteres se convertirán a mayúsculas.

HPFS

El sistema de archivos HPFS se presentó por primera vez con OS/2 1.2 para permitir un mejor acceso a los discos duros de mayor tamaño que estaban apareciendo en el mercado. Además, era necesario que un nuevo sistema de archivos ampliara el sistema de nomenclatura, la organización y la seguridad para las crecientes demandas del mercado de servidores de red. HPFS mantiene la organización de directorio de FAT, pero agrega la ordenación automática del directorio basada en nombres de archivo. Los nombres de archivo se amplían hasta 254 caracteres de doble byte. HPFS también permite crear un archivo de "datos" y atributos especiales para permitir una mayor flexibilidad en términos de compatibilidad con otras convenciones de nomenclatura y seguridad. Además, la unidad de asignación cambia de clústeres a sectores físicos (512 bytes), lo que reduce el espacio perdido en el disco.

En HPFS, las entradas de directorio contienen más información que en FAT. Además del archivo de atributos, esto incluye información sobre la fecha y la hora de modificación, de creación y de acceso. En lugar de señalar al primer clúster del archivo, en HPFS las entradas del directorio señalan a FNODE. FNODE puede contener los datos del archivo, o bien punteros que pueden señalar a datos del archivo o a otras estructuras que, a su vez,

señalarán a datos del archivo.

HPFS intenta asignar, en la medida de lo posible, la mayor cantidad de datos de un archivo en sectores contiguos. Esto se hace con el fin de aumentar la velocidad al realizar el procesamiento secuencial de un archivo.

HPFS organiza una unidad en una serie de bandas de 8 MB y, siempre que sea posible, un archivo estará contenido dentro de una de estas bandas. Entre cada una de estas bandas hay mapas de bits de asignación de 2 KB, que hacen un seguimiento de los sectores dentro de una banda que se han asignado y que no se han asignado. La creación de bandas aumenta el rendimiento porque el cabezal de la unidad no tiene que volver a la parte superior lógica (normalmente el cilindro 0) del disco, sino al mapa de bits de asignación de banda más cercano, para determinar dónde se almacenará un archivo.

Además, HPFS incluye un par de objetos de datos especiales únicos:

Superbloque

El superbloque se encuentra en el sector lógico 16 y contiene un puntero al FNODE del directorio raíz. Uno de los mayores peligros de usar HPFS es que si el superbloque se pierde o resulta dañado debido a un sector defectuoso, lo mismo ocurrirá con el contenido de la partición, incluso aunque el resto de la unidad esté bien. Sería posible recuperar los datos de la unidad copiando todo a otra unidad con un sector 16 en buen estado y volviendo a generar el superbloque. Sin embargo, es una tarea muy compleja.

Bloque de reserva

El bloque de reserva se encuentra en el sector lógico 17, y contiene una tabla de "revisiones" y el bloque de directorio de reserva. En HPFS, cuando se detecta un sector defectuoso, la entrada de las "revisiones" se usa para señalar lógicamente a un sector en buen estado existente en lugar de al sector defectuoso. Esta técnica para el tratamiento de errores de escritura se conoce como revisión.

La revisión es una técnica en la que, si se produce un error debido a un sector defectuoso, el sistema de archivos mueve la información a otro sector diferente y marca el sector original como no válido. Todo ello se realiza de forma transparente para cualquier aplicación que esté realizando operaciones de E/S de disco (es decir, la aplicación nunca sabe que hubo problemas con el disco duro).

Procesos de Microsoft Windows

Todo sistema operativo basa su funcionamiento en una correcta ejecución de los procesos que lo componen. Estos procesos corresponden, entre otras cosas, a las aplicaciones que en cada momento ejecuta el usuario. Sin embargo, además de los asociados al navegador o al juego de turno, nos encontramos con otros llamados "de sistema", que sostienen el funcionamiento general de Windows.

Si quieres saber qué procesos se están ejecutando en este momento en tu equipo, basta con visitar al Administrador de tareas de Windows, pulsando la combinación de teclas Ctrl + Alt + Supr y acudir a la opción correspondiente.

Hay un proceso del sistema muy conocido por los usuarios de Windows, se trata del famoso svchost.exe. Según Microsoft, svchost.exe es el nombre genérico de un proceso anfitrión para servicios que se ejecutan a través de librerías enlazadas dinámicamente. Hace tiempo que la compañía migró las funciones de los servicios de Windows a ficheros DLL en lugar de EXE. Pero, como no se puede ejecutar un archivo DLL directamente en Windows, tiene que ser cargado con un ejecutable, y precisamente svchost.exe fue ese archivo.

Normalmente hay varias instancias de “svchost” corriendo simultáneamente. No es adecuado ejecutar todos los servicios desde una única, ya que un fallo en la misma haría

que todos los servicios asociados se colgasen. De ahí que sea conveniente utilizar un proceso svchost.exe para cada servicio. Por eso tienes tantos procesos iguales en el administrador de tareas.

Entre estos procesos mencionados anteriormente, también te puedes encontrar con otros que a simple vista tal vez no te suenen, pero que son igualmente importantes. Algunos ejemplos:

dwm.exe - Este proceso corresponde a uno de los componentes de Windows más importantes a nivel visual, el llamado Desktop Window Manager. Se trata del gestor de composición que ofrece la capacidad de que disfrutemos de esos bonitos efectos estéticos como ventanas transparentes, miniaturas en vivo en la barra de tareas o el intercambiador de tareas.

ctfmon.exe - Este proceso controla la entrada de usuario alternativa y la barra de idiomas de Office. De este modo, podremos manejar nuestro ordenador a través de la voz o de un puntero y una tableta gráfica, además de caracteres mediante el teclado virtual.

rundll32.exe o rundll.exe - Como no hay una forma directa de ejecutar ficheros DLL, este proceso se encarga de ejecutar las funcionalidades que se almacenan en archivos DLL compartidos. Este ejecutable es una parte válida de Windows, por lo que no representa un problema en la mayoría de las ocasiones.

wuauclt.exe - Es el actualizador de Windows y funciona en segundo plano. Descarga parches y los instala siempre y cuando lo tengas configurado para que detecte automáticamente actualizaciones y las instale por ti.

Wmpnscfg.exe - Windows Media Player puede compartir contenidos multimedia entre ordenadores de la misma red y también con la XBOX 360. Este proceso comparte la biblioteca multimedia incluso aunque no esté abierta. Si sacas provecho de esta

funcionalidad. estás ante una de las grandes características de Windows.

csrss.exe - Conocido como "Client/Server Runtime Subsystem" es un componente que proporciona el modo usuario del subsistema de Windows y que se complementa con win32.sys, que se encarga de la parte que se ejecuta en modo núcleo. Este proceso no debe detenerse bajo ningún concepto, ya que dependen de él las ventanas de consola, la creación y borrado de hilos, y de algunas porciones del entorno virtual de MS-DOS.

mdm.exe - Forma parte de Visual Studio .Net (Entorno de desarrollo para crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET).

services.exe - Es el proceso que controla y gestiona los servicios de Windows.

Administración de la Memoria en Windows

En los sistemas operativos modernos como Windows, las aplicaciones y muchos procesos de sistema *siempre* hacen referencia a la memoria a través de direcciones de memoria virtual. El hardware traduce las direcciones de memoria virtual automáticamente a direcciones reales (RAM). Solo las partes principales del kernel del sistema operativo omiten esta traducción de direcciones y usan directamente las direcciones de memoria reales.

La memoria virtual siempre se usa, incluso cuando la memoria necesaria para todos los procesos en ejecución no supera el volumen de RAM que se instala en el sistema.

Procesos y espacios de direcciones

A todos los procesos (por ejemplo, los ejecutables de aplicaciones) que se ejecutan en versiones de 32 bits de Windows se les asignan direcciones de memorias virtuales (un *espacio de dirección virtual*) que van de 0 a 4 294 967 295 (2 * 32 - 1 = 4 GB), sin importar la cantidad real de RAM que se instala en el equipo.

En la configuración predeterminada de Windows, 2 GB de este espacio de direcciones virtuales se asignan al uso privado de cada proceso, mientras que los 2 GB restantes se comparten entre todos los procesos y el sistema operativo. Generalmente, las aplicaciones (por ejemplo, Bloc de notas, Word, Excel y Acrobat Reader) usan solo una fracción de los 2 GB de espacio de direcciones privadas. El sistema operativo asigna marcos de página de RAM solo a las páginas de memorias virtuales que están en uso.

La extensión de dirección física (PAE) es la característica de la arquitectura de 32 bits de Intel que expande la dirección de memoria física (RAM) a 36 bits. La PAE no cambia el tamaño del espacio de direcciones virtuales (que sigue siendo de 4 GB), sino el volumen de la RAM real que puede admitir el procesador.

Archivo de paginación

La memoria RAM es un recurso limitado, mientras que, a efectos prácticos, la memoria virtual es infinita. Pueden existir varios procesos, cada uno con sus 2 GB de espacio de direcciones virtuales privadas. Cuando la cantidad de memoria que usan todos los procesos existentes supera la RAM disponible, el sistema operativo mueve las páginas (piezas de 4 KB) de uno o más espacios de direcciones virtuales a la unidad de disco del equipo. Esto libera ese marco de RAM para darle otros usos. En los sistemas Windows, estas páginas "paginadas" se almacenan en uno o más archivos (archivos Pagefile.sys) en la raíz de una partición. Puede haber solo un archivo de estos en cada partición de disco. La ubicación y el tamaño del archivo de paginación están configurados en **Propiedades del sistema**.

Los usuarios suelen preguntar "¿qué tamaño debe tener el archivo de paginación?". No hay una respuesta para esto, porque depende de la cantidad de memoria RAM instalada y la memoria virtual que requiere la carga de trabajo. Si no hay otra información disponible, un buen punto de partida sería usar la cantidad generalmente recomendada de 1,5 veces la memoria RAM instalada. En los sistemas de servidor, suele ser recomendable contar con suficiente memoria RAM para que nunca falte y para que el archivo de paginación prácticamente no se use. En estos sistemas, no tiene sentido mantener un archivo de paginación muy grande. Por otro lado, si el espacio en disco es abundante, un archivo de paginación grande (por ejemplo, 1,5 veces la memoria RAM instalada) no generaría problemas y se evitaría la necesidad de preocuparse por su tamaño.

Rendimiento, límites arquitectónicos y RAM

En cualquier equipo, cuanto mayor sea la carga (la cantidad de usuarios y el volumen de trabajo), menor será el rendimiento, pero de forma no lineal. Cualquier aumento en la carga o en la demanda que supere ese punto hará mella de forma considerable en el rendimiento. Esto significa que algún recurso empieza a ser insuficiente y se convierte en un cuello de botella.

En algún punto, el recurso que es insuficiente no puede aumentarse. Esto significa que se ha alcanzado un *límite arquitectónico*. Algunos de los límites arquitectónicos más informados en Windows son los siguientes:

- 2 GB de espacio de direcciones virtuales compartido para el sistema (kernel)
- 2 GB de espacio de direcciones virtuales privado por proceso (modo de usuario)
- 660 MB de almacenamiento de PTE del sistema (Windows Server 2003 y versiones anteriores)

- 470 MB de almacenamiento del bloque paginado (Windows Server 2003 y versiones anteriores)
- 256 MB de almacenamiento del bloque no paginado (Windows Server 2003 y versiones anteriores)

Esto se aplica específicamente a Windows Server 2003, pero también puede aplicarse a Windows XP y Windows 2000. Sin embargo, Windows Vista, Windows Server 2008 y Windows 7 no comparten todos estos límites arquitectónicos. Los límites de memoria del usuario y del kernel (números 1 y 2 en este documento) son los mismos, pero los recursos del kernel, como los PTE y los bloques de memoria, son dinámicos. Estas nuevas características permiten el uso de la memoria paginada y no paginada. Esto también permite que los PTE y el grupo de sesiones superen los límites de los que se habló antes, hasta el punto en que se alcance la máxima capacidad del kernel.

Con frecuencia encontramos las siguientes afirmaciones:
Con un servidor de Terminal Server, los 2 GB de espacio de direcciones compartido se usarán por completo antes de que se utilicen los 4 GB de memoria RAM."

Esto puede ocurrir en algunos casos. Sin embargo, debe supervisar su sistema para saber si esto se aplica o no a su caso. Ciertas veces, estas afirmaciones son conclusiones de entornos específicos de Windows NT 4.0 o Windows 2000, y no necesariamente se aplican a Windows Server 2003. Se realizaron cambios significativos en Windows Server 2003 para reducir la probabilidad de que se alcancen estos límites arquitectónicos en la práctica. Por ejemplo, algunos procesos que tenían lugar en el kernel se sacaron de allí para usar menos memoria en el espacio de direcciones virtuales compartido.

Supervisión del uso de la memoria RAM y la memoria virtual

El Monitor de rendimiento es la herramienta principal para supervisar el rendimiento del

sistema e identificar la ubicación del cuello de botella. Para iniciar el Monitor de rendimiento, haga clic en **Inicio**, **Panel de control**, **Herramientas administrativas** y, después, doble clic en **Monitor de rendimiento**. A continuación, encontrará un resumen de contadores importantes y lo que indican:

- Memoria, Bytes confirmados: este contador es una medida de la demanda de la memoria virtual.

  Aquí se indica la cantidad de bytes asignados por proceso y para qué proceso el sistema operativo confirma un marco de página de RAM o un espacio de página en el archivo de paginación (o quizás ambos). Cuando los **bytes confirmados** superan la memoria RAM disponible, la paginación aumenta, al igual que lo hace el tamaño del archivo de paginación que se usa. En algún punto, la actividad de paginación empieza a afectar el rendimiento de forma considerable.

- Proceso, espacio de trabajo, _Total: este contador es una medida de la memoria virtual en uso "activo".

  Este contador muestra la cantidad de memoria RAM requerida para que la memoria virtual que se está usando para todos los procesos esté en la memoria RAM. Este valor siempre es un múltiplo de 4096, que es el tamaño de página que se utiliza en Windows. Cuando la demanda de memoria virtual supera la memoria RAM disponible, el sistema operativo ajusta la cantidad de memoria virtual de un proceso asignada a su espacio de trabajo para optimizar el uso de RAM disponible y minimizar la paginación.

- Archivo de paginación, % del archivo de paginación en uso: este contador es una medida del total utilizado del archivo de paginación.

  Use este contador para determinar si el archivo de paginación tiene un tamaño adecuado. Si este contador llega a 100, significa que el archivo de paginación está completo y todo deja de funcionar. Según la volatilidad de la carga de trabajo,

probablemente sea conveniente que el archivo de paginación sea lo suficientemente grande como para que no se utilice más del 50 al 75 %. Si se utiliza gran parte del archivo de paginación, es probable que el rendimiento aumente si se cuenta con más de uno en los diferentes discos físicos.

- Memoria, Páginas/s: este contador ofrece una de las medidas que más se malinterpretan.

  Si el contador muestra un valor alto, esto no quiere decir necesariamente que el cuello de botella de rendimiento se deba a una falta de memoria RAM. El sistema operativo usa el sistema de paginación para otros fines aparte del intercambio de páginas debido a la asignación excesiva de memoria.

- Memoria, Salida de páginas/s: este contador muestra la cantidad de páginas de memoria virtual que se escribieron en el archivo de paginación por segundo para liberar marcos de página de RAM para otros fines.

  Es el mejor contador para hacer un control si sospechas que la paginación es el cuello de botella de rendimiento. Aunque el valor de Bytes confirmados supere la memoria RAM y el de Salida de páginas/s sea inferior o igual a cero la mayor parte del tiempo, no existe un problema de rendimiento importante derivado de tener poca memoria RAM.

- Memoria, Bytes confirmados,
  Memoria, Bytes de bloque no paginado,
  Memoria, Bytes de bloque paginado,
  Memoria, Total de bytes de código del sistema,
  Memoria, Total de bytes de controladores del sistema:
  La suma de estos contadores indica qué cantidad se está usando realmente de los 2 GB correspondientes a la parte compartida del espacio de direcciones virtuales de 4 GB. Usa este dato para determinar si el sistema está llegando a uno de los límites arquitectónicos de los que se habló antes.

- Memoria, MBytes disponibles: este contador mide la cantidad de RAM disponible para satisfacer demandas de memoria virtual (ya sea nuevas asignaciones o a fin de restaurar una página a partir de un archivo de paginación).

  Cuando la memoria RAM escasea (por ejemplo, si los bytes confirmados superan la memoria RAM instalada), el sistema operativo intenta mantener disponible una cierta fracción de la memoria RAM instalada para uso inmediato. Para ello, copia páginas de memoria virtual que no están en uso activo al archivo de paginación. Por lo tanto, este contador no llega a cero y no indica necesariamente si al sistema le falta RAM.

Conclusiones y recomendaciones

Windows es un sistema que aprovecha la potencia de los procesadores, ha sido diseñado para adaptarse a las nuevas tecnologías, ofrece compatibilidad con varias plataformas (OS/2, Unix y versiones anteriores a el mismo), soporta el multiprocesamiento simétrico, buen rendimiento y conectividad, seguridad y al no estar encasillado en ningún modelo estandar de Sistema Operativo tiene la capacidad de combinar las ventajas del modelo cliente/servidor, puede correr además sobre múltiples arquitecturas con un mínimo de cambios, permite que varios procesos sean ejecutados simultáneamente en varios procesadores y estos no se apropien de recursos del sistema por tiempo indefinido, sino por tratamiento del sistema.

Referencias bibliográficas

Delan Han. (2020) Introducción a los archivos de pagina, direccion de extraccion (https://docs.microsoft.com/es-es/windows/client-management/introduction-page-file)

UPSE. (2015) Kernel de los sistemas operativos, direccion de extraccion (https://es.slideshare.net/XlxAcost/kernel-de-los-sistemas-operativos)

EcuRed. (2018) Kernel, direccion de extraccion (https://www.ecured.cu/Kernel)

San Marcos

www.usanmarcos.ac.cr

San José, Costa Rica