

4. Funciones y características de los Sistemas Operativos

En el apartado anterior, al hablar de los tipos de sistemas operativos se han señalado algunas de sus características y rasgos deseables. Además, por medio del desarrollo histórico, se ha visto la necesidad de que el sistema operativo proporcione una serie de funciones. En este punto se va a hablar de ellas más detenidamente, refiriéndose sobre todo a los sistemas de propósito general, ya que engloban la mayoría de las funciones y características de los diferentes sistemas operativos.

Los sistemas de propósito general se caracterizan por tener un gran número de usuarios trabajando sobre un amplio abanico de aplicaciones. Están diseñados para mantener un flujo constante de trabajo en forma de tareas a ejecutar. Como las tareas a realizar serán muy diversas, deberán soportar un gran número de utilidades (compiladores, editores, ensambladores, paquetes de depuración, procesadores de texto, sistemas de ficheros para almacenamiento a largo plazo, etc ...) y de periféricos. Su control y disponibilidad junto con la organización del flujo de trabajo es su principal función. En estos sistemas se combinan sistemas 'batch' (locales o remote job entry) y sistemas de acceso múltiple (el usuario desde un terminal lanza, dirige y controla la ejecución), aprovechando lo primero para actividades rutinarias no interactivas.

Un sistema operativo crea el entorno en el que se ejecutan los programas, ofreciendo a los programas y a los usuarios una serie de servicios. Antes de comenzar el diseño del sistema, sus objetivos deberán estar bien definidos ya que serán la base de toma de decisiones de diseño, tales como la elección de algoritmos y estrategias.

La construcción de sistemas operativos modernos requiere la división de los mismos en diferentes partes con funciones claramente diferenciadas y con interfaces de entrada y salida adecuados. Los servicios específicos ofrecidos por los sistemas operativos varían de unos a otros, pero existen algunas clases de servicios que son comunes y que pueden identificarse claramente.

Gestión de procesos

El sistema realiza varias actividades simultáneamente que compiten por la utilización de determinados recursos, entre los que destaca la CPU, estas actividades se denominan procesos. Un programa, por si solo, es una entidad pasiva mientras que un proceso es una entidad activa ya que compite por los recursos máquina. El proceso es una unidad de trabajo del sistema y el sistema operativo es el responsable de realizar un seguimiento de todas las actividades del sistema en conexión con los procesos

generados. Para ello crea y destruye procesos, suspende y reanuda su ejecución, ofrece los mecanismos de sincronización y comunicación y permite la espera durante la ejecución de un proceso.

Gestión de memoria

Existen diversos esquemas para gestionar la memoria de un sistema, la cual es vital para el funcionamiento del mismo. Generalmente, la elección de uno u otro esquema depende de las posibilidades hardware.

El sistema operativo es por tanto responsable de registrar el uso que se hace de una determinada parte de memoria, decidir qué procesos deben ser cargados en memoria para ejecución y ser capaz de ocupar o liberar el espacio de memoria que utilizan los procesos.

Gestión del almacenamiento

La memoria real instalada suele ser insuficiente para acomodar todos los datos y programas permanentemente; el sistema debe proporcionar mecanismos de gestión de almacenamiento secundario para soportar este almacenamiento. El sistema operativo es responsable de gestionar el espacio libre en los dispositivos, localizar los espacios de almacenamiento y planificar la utilización del disco.

Gestión de entradas y salidas

El sistema de entradas y salidas consiste en un sistema de 'buffering', los drivers e interfaces necesarios para los dispositivos (solamente los drivers conocen las características específicas de los dispositivos).

Gestión de ficheros

El sistema operativo ofrece una visión uniforme de todo el espacio de almacenamiento disponible abstrayendo las características físicas de cada dispositivo y definiendo una unidad lógica de almacenamiento que es el fichero. El sistema operativo es por tanto el responsable de la creación y borrado de ficheros y directorios, de proporcionar las primitivas necesarias para su utilización y de 'mapear' los ficheros sobre el espacio de almacenamiento.

Gestión de protección

La protección incluye los mecanismos que controlan el acceso de los programas, procesos y usuarios a los recursos del sistema. Todos los recursos del sistema deben ser manipulados solamente por aquellos procesos autorizados por el sistema operativo. Se

debe asegurar que cada proceso sólo utilice su espacio de direcciones, y tenga el control de la CPU un espacio de tiempo determinado.

Comunicaciones

Los distintos procesadores existentes en un sistema distribuido, deben conectarse mediante redes de telecomunicación que se pueden configurar de diferentes maneras. El sistema debe ofrecer al usuario acceso a los diferentes recursos pese a su distribución geográfica; habitualmente se generaliza el acceso a la red como un acceso a un tipo de fichero especial. La gestión de las comunicaciones incluye el 'enrutamiento', las estrategias de conexión y las soluciones a problemas de contención y seguridad.

Intérprete de comandos

Es uno de las partes más importantes de los sistemas operativos actuales, y proporciona la posibilidad de comunicación, mediante comandos, con el sistema operativo; su función consiste en obtener una nueva orden, interpretarla y atenderla.

4.1 Funciones de los Sistemas Operativos

Funciones dirigidas al usuario

Para el usuario las funciones o servicios del sistema operativo están encaminadas a transformar el 'hardware' instalado (RCP o poder de cálculo bruto), en una máquina más tratable, más fácil de usar, proporcionando una máquina virtual con características aparentemente diferentes de la máquina física que hay detrás.

Las áreas en las que más se diferencia la máquina instalada del sistema que se presenta al usuario son:

- **Ambiente de ejecución:**
 - **Ejecución y control de programas:** Los usuarios desearán ejecutar programas. El sistema debe ser capaz de cargar en memoria un programa y ejecutarlo. El programa podrá acabar su ejecución de manera normal o anormal caso de algún error. Además, los sistemas operativos deben proporcionar a los usuarios un lenguaje de comandos que les permita transmitir a la máquina sus peticiones: compilar, montar, ejecutar un programa, etc...
 - **Uso de la memoria:** Muchos sistemas operativos presentan al usuario una memoria que difiere en tamaño de la real instalada. Así el sistema

operativo ofrece a cada usuario una cantidad de memoria adecuada para sus actividades.

- **Comunicaciones:** Existen diversas circunstancias en las que un proceso debe comunicarse con otro. Esta comunicación se realizará a través de un red de telecomunicaciones, o simplemente mediante la compartición de memoria o el paso de mensajes.
- **Operaciones de Entrada/Salida:** La utilización de los mecanismos de E/S proporcionados por el 'hardware' básico, son extremadamente complejos y requieren una programación sofisticada. El sistema operativo libera al usuario de tener que comprender esta complejidad presentándole unas posibilidades de E/S sobre ficheros o dispositivos, igual de potentes pero mucho más sencillas de manejar.
- **Manipulación de ficheros:** Los sistemas de ficheros se basan en el almacenamiento en dispositivos como discos, cintas, etc... Los sistemas operativos permiten al usuario acceder a la información en ellos almacenada, mediante nombres simbólicos en lugar de por su posición física en el medio de almacenamiento.

A través del sistema de nombres un usuario puede crear y borrar un fichero y realizar con él operaciones de lectura y escritura. La misión del sistema operativo es facilitar los accesos y mantener los ficheros íntegros y sin errores.

- **Detección y gestión de errores:** El sistema operativo debe detectar cualquier posible error. Los errores pueden producirse en la CPU y en el hardware de memoria (error por fallos electromecánicos), en un dispositivo (error de paridad, por ejemplo) o bien en un programa de usuario (como un 'overflow' aritmético o una tentativa ilegal de acceso a memoria). Para cada tipo de error el sistema operativo debe adoptar una iniciativa apropiada que garantice un proceso correcto y consistente.

Funciones internas del Sistema Operativo

Estas funciones del sistema operativo están dirigidas al usuario, pero existen otras funciones que el sistema debe soportar y que permiten un funcionamiento eficiente del sistema. Los sistemas operativos, generalmente deben soportar la ejecución simultánea o paralela de varias actividades. Es decir, deben dar soporte a la concurrencia.

Para ello el sistema debe implementar otro conjunto de funciones que permitan, principalmente, la compartición de recursos. Esta compartición es necesaria por las razones siguientes:

- **Coste:** es impensable proporcionar recursos suficientes de forma separada para todos los usuarios;
- **Trabajo a partir de otros trabajos ya hechos** usando rutinas, programas o trozos de ello;
- **Existencia de datos compartidos:** programas y usuarios diferentes pueden necesitar usar la misma información, por ejemplo sita en una misma base de datos;
- **Eliminación de redundancia:** por ejemplo, usando una sola copia de un programa para todos los usuarios.

Para soportar la concurrencia el sistema operativo debe implementar las siguientes funciones:

- **Asignación de recursos:** Cuando existen varios usuarios ejecutando diferentes tareas, los recursos del sistema deben asignarse a cada tarea intentando siempre aumentar la disponibilidad de la máquina y así mismo optimizar el uso de los recursos. El sistema operativo gestiona muchos tipos diferentes de recursos. Para algunos de ellos (como la CPU, memoria principal o almacenamiento en ficheros), hay que disponer de programas especiales de asignación mientras que para otros (como los dispositivos) basta con programas de solicitud y liberación mucho más generales.
- **System Accounting:** En los sistemas donde trabajan varios usuarios es deseable controlar los pasos de los usuarios que utilizan el ordenador, en qué medida lo hacen y qué tipo de recursos emplean. Esta información puede emplearse a la hora de la facturación de servicios, o simplemente para la realización de estadísticas. Estas estadísticas, son una herramienta muy valiosa para optimizar el sistema (tuning), quizás reconfigurándolo, de forma que se adapte mejor a los usuarios y sus actividades.
- **Protección:** Los propietarios de la información almacenada en un sistema informático compartido deben poder controlar el uso que se haga de dicha información. Además los trabajos que están ejecutándose simultáneamente no deben interferir unos con otros. Todas las demandas concurrentes de

varios recursos deben ser conciliadas equitativamente y programadas razonablemente.

4.2 Componentes de los Sistemas Operativos

Hay muchas formas de dividir un sistema operativo en bloques funcionales, pero todas tienen cabida en el esquema de división mostrado en la figura 1.x. Según esta división, un sistema operativo tiene dos componentes: Componente gestor (supervisor) y Componente soporte (soporte de servicios).

Componente Gestor	Gestor de trabajos		Gestor de Tareas	Gestor de datos
	Función Soporte de la Gestión			
Componente Soporte	Ayudas al mantenimiento	Compilador		Utilidades
	De prog. de aplicación	Intérprete		Cargador

Figura 1.9: Componentes de un sistema operativo

Componente Gestor

Este componente es responsable de regular las actividades del sistema de proceso de datos y está formado por el conjunto de funciones del sistema operativo que controla dinámicamente el ambiente total del sistema de proceso de datos. Estas funciones son realizadas por:

- El gestor de trabajos o 'jobs'¹ es el encargado de la preparación de los trabajos antes de su ejecución en el procesador central asignándoles los recursos que van a necesitar. Estos recursos son básicamente indicados por el usuario mediante el lenguaje de control del sistema operativo u OSCL².

1 Job o trabajo: Unidad mínima de trabajo que puede ser presentada al ordenador para su ejecución. Se divide en pasos de trabajo. Esta división lógica, permite la interrupción de un job en caso de error.

2 OSCL o lenguaje de control del sistema operativo: Es el medio relevante mediante el que el usuario puede pedir recursos al sistema para la ejecución de programas. Además debe permitir:

Por ejemplo, memoria principal, ficheros, programas a ejecutarse, posición en la cola de trabajos³, dispositivos, volúmenes de almacenamiento auxiliar, prioridad de ejecución.

- El gestor de tareas que supervisa dinámicamente de los recursos que utiliza cada tarea⁴, diferentes de E/S, cuya necesidad no se puede prever o conocer antes de su creación. Por ejemplo memoria principal complementaria, tiempo de UCP, recuperación tras errores, etc...
- El gestor de datos cuyo cometido consiste en la asignación de memoria auxiliar, la supervisión de todas las rutinas y actividades de E/S incluidos los interfaces de usuario, la gestión de espacio, los métodos de acceso⁵, programas de canal⁶, la gestión de 'buffers'⁷ y la recuperación tras errores.

Componente Soporte

El componente soporte, se encarga de asegurar que las actividades del sistema de proceso de datos puedan tener lugar, ayudando al usuario a mantener los datos, el sistema, los programas y la instalación; en consecuencia, este componente podría definirse como componente de mantenimiento del sistema operativo y de ayuda al usuario: su soporte. Las funciones del sistema operativo que incluye son las que mantienen y soportan los programas y datos del usuario y del propio sistema operativo, estas son:

- Soporte de Gestión que permite llevar un registro de los recursos empleados por los trabajos individuales y para ello proporciona programas de medida del

-
- Dar un nombre único a la unidad de trabajo presentada.
 - Especificar los programas (en particular) a ejecutarse en cada paso de trabajo.
 - Definir las unidades de E/S y ficheros necesarios.

Las rutinas de gestión de trabajos leen e interpretan OSCL y realizan la asignación de recursos.

3 Cola de trabajos: Área en el almacenamiento externo que contiene registros de los trabajos presentados al ordenador (con OSCL no interactivo).

4 TAREA o proceso: Es la unidad más pequeña de trabajo que puede competir activamente por recursos del sistema; es una unidad de trabajo interna, creada por el sistema operativo al aceptar un paso de trabajo.

5 Los métodos de acceso son una parte esencial del sistema de ficheros; construyen programas de canal estándar y permiten realizar la supervisión dinámica de todas las actividades de E/S.

6 Los programas de canal son una secuencia de mandatos que se envían como una unidad al controlador de dispositivos para la realización de las actividades de E/S.

7 Un buffer es un área de memoria para guardar temporalmente la información a introducir o sacar de memoria principal mediante una operación de E/S.

sistema, simuladores del sistema, rutinas de contabilidad, utilidades de gestión especial de memoria principal y secundaria, programas 'dump' para volcar la memoria, rutinas de 'check-point / restart', etc.; es decir, incluye todo lo necesario para poner a punto el sistema ('tuning'), de forma que su rendimiento sea óptimo.

- Ayudas para el mantenimiento de programas de aplicación que son un conjunto de rutinas que facilitan al usuario el mantenimiento de sus aplicaciones; es decir, posibilitan la utilización de bibliotecas, proporcionan generadores de módulos de carga, utilidades para modificar instrucciones de programas en memoria principal y auxiliar, etc...
- Compiladores e Intérpretes que proporcionan la capacidad de utilización de lenguajes de alto nivel, más cercanos al lenguaje humano. son programas producto que incluyen además del traductor, bibliotecas con módulos fuente para la fase de compilación y bibliotecas con módulos objeto para la fase de montaje.
- Utilidades y Cargadores que facilitan las tareas rutinarias como por ejemplo: formatear, catalogar, generar listados, editores, copiar, mover, borrar ficheros, cargar, intercalar, clasificar, etc.

4.3 Acceso a los servicios del sistema

Los sistemas operativos ofrecen sus servicios de diferentes maneras pero existen dos modos básicos de acceso a dichos servicios por parte del sistema operativo: *las llamadas al sistema y los programas del sistema*.

Llamadas al sistema

Sirven para obtener los servicios de más bajo nivel, constituyen el interfaz entre los programas en ejecución y el sistema operativo. Generalmente estas llamadas están implementadas en instrucciones en ensamblador. Algunos lenguajes de alto nivel, como C, permiten realizar llamadas al sistema directamente, otros, como el Pascal, las realizan a través de rutinas que al ejecutarse llaman al sistema ('run-time routines').

Cuando se realiza una llamada al sistema es necesario, a parte de su identificación, una serie de informaciones necesarias durante su ejecución. Estas informaciones pueden pasarse a la rutina de servicio llamada mediante tres métodos. La forma más simple es depositar los datos en registros máquina, pero a veces el número o longitud de los mismos supera el tamaño de los registros, en estos casos se almacenan

en tablas de memoria y en los registros se depositan las direcciones de esas tablas. También se puede utilizar la pila del sistema, en la que los programas introducen los parámetros y la rutina extrae las posiciones correspondientes.

Las llamadas al sistema pueden agruparse en cinco categorías:

- Manipulación de archivos: Estas llamadas al sistema permiten crear y borrar un fichero (create y delete). Para poder manejarlo correctamente deberemos abrirlo y cerrarlo (open y close) y por supuesto deberemos poder escribir o leer dicho fichero (read y write). En algunos casos es necesario reposicionarse dentro del espacio de un fichero, bien colocándose al principio, al final o en algún punto conocido (reposition).

Si el sistema dispone de una estructura de directorios, puesto que son archivos lógicos se usan para el correcto mantenimiento de la información, las mismas llamadas pero aplicadas a los directorios. Además muchas veces es necesario conocer o manipular los atributos de un fichero o directorio (get y set file attribute)

- Control de procesos y trabajos: Permiten la terminación de programas, bien de modo normal (end) o anormalmente (abort). Generalmente se trabaja con niveles de error siendo los números más altos indicativos de errores más severos, así una terminación normal correspondería a un nivel de error 0. Estos niveles de error permiten al sistema secuenciar las tareas y poder recuperarse en caso de error.

Es posible que un proceso en ejecución requiera cargar y ejecutar (load y execute) otro proceso. Cuando esto ocurre, existen varias posibilidades para el proceso llamador: continuar su ejecución de forma paralela al nuevo proceso (create process o submit job), esperar durante un cierto tiempo, a la ejecución del nuevo proceso (wait time); esperar a que se produzca un suceso determinado (wait event) que deberá ser señalado por el nuevo proceso (signal event). Generalmente se desea controlar la ejecución de los nuevos procesos a través de sus atributos (get y set process attributes) o terminar su ejecución (terminate process).

Otro conjunto de llamadas útiles son las que permiten la depuración de un programa, mediante vaciados de memoria (dump) o mediante su ejecución paso a paso (trace). Además algunos sistemas ofrecen información correspondiente al perfil temporal de un programa.

- Manipulación de dispositivos: Generalmente los ficheros pueden considerarse una especie de dispositivos virtuales. Las llamadas al sistema para manejo de los dispositivos pueden ser las mismas que para el manejo de ficheros. Además si se trabaja en un sistema multiusuario, es necesario solicitar y liberar los dispositivos (request y release) y en algunos casos posiblemente reposicionarse (reposition). La mayoría de sistemas combinan una estructura mixta de ficheros y dispositivos. En este caso los dispositivos son ficheros con nombre y atributos especiales.
- Mantenimiento de información: Son las llamadas al sistema que permiten la transferencia de información entre el programa de usuario y el sistema operativo. Por ejemplo un programa puede obtener la fecha y hora del sistema (date y time). Se puede obtener información del propio sistema operativo, como su versión, o su fecha de creación.

Además se puede obtener información sobre el estado del sistema, tamaño de memoria libre, número de usuarios conectados, el espacio disponible en disco, etc. Por otro lado, el sistema mantiene información sobre todos los procesos que se encuentran en ejecución, existiendo llamadas para poder acceder a esta información.

- Comunicaciones: Principalmente existen dos sistemas o modelos de comunicación. En el sistema de mensajes, la información es transferida por el sistema operativo de un proceso a otro utilizándose varias llamadas al sistema; es decir, el sistema operativo se encarga de resolver todos los problemas de comunicación.

Así, antes de comenzar la comunicación se debe crear una conexión (open connection), si el sistema es distribuido será necesario obtener los identificativos de la máquina y el proceso con el cual nos queremos comunicar (get hostid y get processid). Una vez abierta la conexión es necesario que el receptor la acepte (accept connection). Para transferir los mensajes utilizaremos las llamadas de lectura o escritura de mensajes (read y write message) y al finalizar cerraremos la conexión (close connection).

El sistema de memoria compartida, en que los procesos usan llamadas al sistema, solamente, para acceder a zonas de memoria compartidas donde la información es depositada y leída. Los problemas de comunicación los debe resolver el programador, el sistema operativo solo proporcionará la memoria a compartir.

Programas del sistema

La mayoría de sistemas operativos modernos son suministrados en dos partes, el código del supervisor del sistema operativo y una amplia colección de programas del sistema para realizar operaciones que permitan crear un entorno más adecuado para el desarrollo y la ejecución de programas de usuario.

Estos programas del sistema se dividen en varias categorías:

- Manipulación de archivos: que permiten crear, borrar, copiar, renombrar, imprimir, y en general manipular ficheros.
- Carga y ejecución de programas: el sistema ofrece los cargadores absolutos o reubicables, los montadores de enlaces y algunos programas para depuración de lenguajes de alto nivel y/o código máquina.
- Soporte de los lenguajes de programación: Facilita los compiladores e interpretes, aunque debido a su diversidad, generalmente estos programas se suministran por separado.
- Comunicaciones: Correo electrónico, etc.
- Modificación de ficheros: Generalmente el sistema operativo ofrece editores de textos con los que modificar el contenido de los ficheros existentes.
- Mantenimiento de información de estado: son programas que permiten obtener información sobre el estado del sistema obteniendo el día, la hora, el número de usuarios conectados, el espacio libre en memoria y en dispositivos, etc. Esta información se obtiene correctamente formateada para su visualización o impresión.
- Programas de aplicación: Son programas añadidos que permiten al usuario resolver algunos problemas comunes, como formatear textos, manejar bases de datos, utilizar gráficos, etc.
- Intérprete de comandos.

Pero, el programa más importante del sistema operativo es el intérprete de comandos. Este programa se ejecuta cuando un usuario se conecta al sistema. Su función consiste en captar los comandos y ejecutarlos.

Estos comandos que en su mayoría trabajan con ficheros, los manipulan, pueden estar implementados de las formas siguientes:

- a) el código de los comandos como parte del intérprete de comandos, lo cual implica que el número y la complejidad de los comandos determina la longitud del intérprete de comandos;
- b) los comandos formando programas especiales del sistema, por lo que el intérprete de comandos no 'entiende' el comando, simplemente carga un programa con un nombre igual al del comando y lanza su ejecución. Esta opción es mucho más adecuada, ya que los sistemas obtenidos son fácilmente ampliables.

Generalmente el código que 'interpreta' el comando y el código del comando no tienen por qué estar en memoria simultáneamente, resultando, el paso de parámetros, una tarea compleja (UNIX). Además la interpretación de los parámetros de cada comando se deja en manos del código del comando, es decir, de su programador.

La idea que el usuario tiene del sistema, es la que le transmiten los programas del sistema, esta visión suele resultar bastante alejada del sistema real. Los problemas de diseño de un interfaz 'friendly' para el usuario son diversos, pero no dependen del sistema operativo. Desde el punto de vista del sistema operativo no hay ninguna distinción entre los programas del sistema y los programas del usuario.

4.4 Características de los Sistemas Operativos

Un sistema operativo de propósito general se caracteriza por tener implementados los conceptos siguientes:

- Facilidad de uso: El sistema operativo debe estar diseñado de la forma más conveniente para los usuarios (iconos, menús,...).
- Generalidad y flexibilidad: Capacidad de adaptación a las diferentes necesidades de los usuarios.
- Visibilidad y opacidad: Hay que mostrar a los usuarios sólo lo que necesiten para sus desarrollos y sus actividades y de la forma en que puedan sacar más rendimiento del sistema.
- Eficiencia y disponibilidad: Rápido, de funcionamiento correcto y proporcionando tantas funciones como sea posible. Suele medirse por diversos criterios como por ejemplo: tiempo medio entre trabajos, tiempo 'muerto' de procesador central, tiempo de ejecución (para procesos 'batch'), tiempo de respuesta (sistemas de acceso múltiple), utilización de recursos,

throughput (número de trabajos / unidad de tiempo), etc.; criterios estos, que no siempre pueden satisfacerse simultáneamente.

- Seguridad e integridad: La protección de los datos, programas de usuario y del propio Sistema Operativo de accesos no autorizados y de posibles daños, es básica.
- Fiabilidad: Lo ideal, y hacia eso se camina, sería que el sistema operativo estuviese libre de error y fuese capaz de responder a cualquier evento o contingencia.
- Determinismo e indeterminismo: Esta claro que un sistema operativo debe ser "determinista" en el sentido de que el mismo programa ejecutado con los mismos datos debe producir los mismos resultados en cualquier momento. Pero al mismo tiempo, debe ser "indeterminista" en el sentido de que debe responder a cualquier suceso imprevisto. Sucesos como por ejemplo peticiones de recursos, errores de programa en tiempo de ejecución e interrupciones de los periféricos. Dado el enorme número de contingencias que pueden darse sería ridículo que el sistema operativo se hiciese para atenderlos de forma individual. En su lugar debe haberse previsto para atender "cualquier" secuencia de eventos.
- Mantenibilidad y extensibilidad: Un sistema operativo tiene que poder ser depurado y ampliado fácil y rápidamente; La construcción modular, con los interfaces entre los módulos claramente definidos y documentados, posibilita su mantenimiento y extensión.
- Tamaño razonable: Para que la disposición de memoria por parte del sistema operativo no afecte a la productividad, su tamaño debe ser razonable; cuanto mayor sea más posibilidades de error y difícil mantenimiento.