



Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica



Sistemas Operativos
Actividad Fundamental #3

Docente: Dra. Norma Edith Marín Martínez

Grupo: 002

Equipo: 2

Nombre	Matricula	Carrera	Semestre
Francisco Gael Reyes Cantú	1995983	IAS	
Frida Jaziry Juarez Fuentes	2028420	IAS	
Isac Alfredo Almaguer Espinosa	2049903	IAS	
Fátima Arizpe Sánchez	2025106	IAS	
Jose Ángel Cardenas Contreras	1935156	IAS	

Día y hora: Sábado M4-M6

Fecha de entrega: 28 de Marzo de 2025

Índice

Introducción.....	3
Tipos de memorias y sistemas de archivos en sistemas operativos.....	4
¿Porque es importante la administración de memoria en estos dispositivos?	18
Funciones clave de la administración de memoria	18
¿Qué debe hacer el sistema operativo para llevar un control de los espacios disponibles y ocupados en la memoria?	21
Técnicas de Administración de Memoria	21
¿Qué problemas son más frecuentes en la administración de memoria?	24
Conclusión	27
Conclusiones individuales	28
Bibliografía	30

Introducción

En el ámbito de los sistemas operativos, el manejo de la memoria y los archivos es fundamental para el rendimiento y la eficiencia de los dispositivos, ya sea en computadoras, dispositivos móviles o servidores de red. La memoria permite que los sistemas operativos ejecuten procesos y almacenen información temporalmente, mientras que los sistemas de archivos proporcionan una estructura para el almacenamiento a largo plazo de datos. Sin una correcta gestión de estos recursos, los sistemas no podrían operar de manera eficiente ni proporcionar una experiencia de usuario óptima.

La administración de memoria es un elemento esencial en el desempeño de los sistemas operativos modernos. Su correcta gestión no solo influye en la velocidad y estabilidad del sistema, sino que también garantiza una distribución eficiente de los recursos, permitiendo la ejecución simultánea de múltiples procesos sin generar conflictos. En los distintos sistemas operativos, como Windows, Linux, macOS, Android, iOS y HarmonyOS, se han desarrollado diversas estrategias para optimizar el uso de la memoria, minimizar problemas como la fragmentación y evitar la sobrecarga de los dispositivos. Este trabajo explora los principales desafíos en la administración de memoria, sus manifestaciones en diferentes entornos operativos y las soluciones implementadas para mitigar estos problemas.

Tipos de memorias y sistemas de archivos en sistemas operativos

Las computadoras utilizan varias formas de memoria, entre ellas:

Memoria Volátil (RAM y Caché)

- **RAM (Memoria de Acceso Aleatorio):** Se usa para ejecutar programas y procesos en tiempo real. Se borra al apagar la computadora.
- **Caché:** Una memoria ultrarrápida dentro del procesador que almacena datos frecuentemente usados para mejorar el rendimiento.

Memoria No Volátil (Almacenamiento)

- **ROM (Memoria de Solo Lectura):** Almacena el firmware o BIOS/UEFI, permitiendo que el sistema arranque.
- **SSD/HDD (Unidades de almacenamiento):** Guardan los sistemas operativos, programas y archivos.
- **Memoria Flash (USB, SSD, Tarjetas SD):** Utiliza chips de memoria NAND para almacenamiento rápido y permanente.

a) Tipos de memorias y sistemas de archivos en computadoras según 3 sistemas operativos

Windows

Tipos de memoria

RAM: Es la memoria principal que se utiliza para ejecutar programas y procesos de manera temporal. Almacena los datos y programas que se están utilizando activamente.

Memoria Virtual: Cuando la RAM se llena, el sistema operativo utiliza una parte del disco duro como extensión de la RAM. Esto se llama memoria virtual y en Windows se implementa mediante un archivo de paginación (pagefile.sys), donde se almacenan datos de procesos inactivos.

Caché de Disco: Windows también utiliza una memoria caché para acelerar el acceso a los datos más frecuentemente utilizados en el disco duro.

Windows maneja la memoria con un esquema basado en **paginación y caché**, además de herramientas para gestionar la memoria virtual.

Sistema de archivos

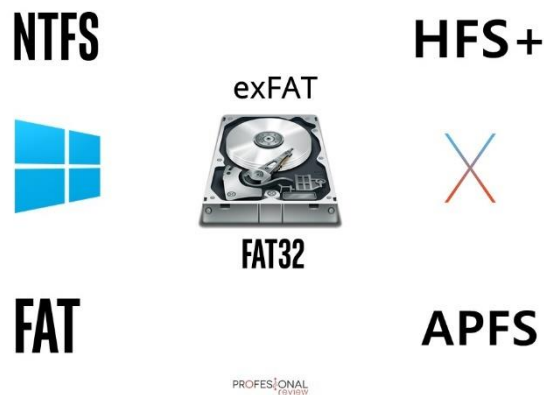
NTFS (New Technology File System): Es el sistema de archivos más avanzado y utilizado por Windows en discos duros y SSD. NTFS ofrece seguridad, permite la compresión de archivos, manejo de permisos de acceso y journaling, que es la capacidad de registrar cambios para evitar corrupción de datos en caso de fallos.

FAT32 (File Allocation Table 32): Compatible con casi todos los sistemas, pero tiene límite de 4GB por archivo.

exFAT (Extended File Allocation Table): Es utilizado mayormente en dispositivos de almacenamiento extraíbles como memorias USB. Es compatible con archivos de gran tamaño, pero no ofrece tantas funciones avanzadas como NTFS, especialmente en seguridad y permisos. Similar a FAT32 pero sin el límite de 4GB, ideal para memorias USB y discos externos.

¿Cómo opera en Windows?

- Usa un administrador de memoria que asigna recursos dinámicamente.
- Utiliza un sistema de paginación (archivo de paginación o swap) en el disco duro para ampliar la memoria virtual.
- Prefiere NTFS por su seguridad y fiabilidad en discos internos.



LINUX

Tipos de memoria

RAM: Al igual que Windows, Linux utiliza RAM para ejecutar programas y procesos activos.

Memoria Virtual (Swap): Cuando la RAM se agota, Linux utiliza un archivo o partición de "swap" para mover los datos de los procesos menos utilizados a la memoria virtual. Esto permite continuar operando, aunque la RAM esté saturada.

Memoria Caché: Linux hace un uso intensivo de la memoria caché para almacenar datos que pueden ser requeridos rápidamente, mejorando el rendimiento.

Linux maneja la memoria de manera muy eficiente, aprovechando **caché agresivo y el uso de swap** solo cuando es necesario.

Sistemas de archivos

ext4 (Fourth Extended File System): Es el sistema de archivos más común en distribuciones de Linux. Soporta grandes volúmenes de datos, journaling y es altamente eficiente en la gestión de espacio y el manejo de errores.

XFS: Es un sistema de archivos de alto rendimiento, ideal para grandes volúmenes y archivos grandes. Se utiliza en servidores y sistemas que requieren un manejo eficiente de archivos masivos.

Btrfs (B-tree File System): Es un sistema de archivos de nueva generación, diseñado para ofrecer funcionalidades avanzadas como snapshots (copias instantáneas), gestión de volúmenes, y recuperación de errores.

SWAP: Espacio en el disco que funciona como extensión de la memoria RAM cuando esta se llena.

¿Cómo opera en Linux?

- Utiliza **swap** para gestionar memoria virtual.
- Implementa técnicas avanzadas de caché y buffers para optimizar el uso del disco.
- Permite cambiar fácilmente el sistema de archivos dependiendo del propósito del dispositivo.



macOS

Tipos de memoria

Memoria RAM (Memoria de Acceso Aleatorio)

- Se usa para cargar y ejecutar aplicaciones activas.
- macOS implementa **Memory Pressure**, un sistema que analiza el uso de la RAM y optimiza su distribución.

Memoria Virtual (Virtual Memory)

- Cuando la RAM se llena, macOS usa parte del almacenamiento (SSD o HDD) como memoria virtual.
- Se gestiona a través de un **archivo de paginación (swapfile)** en el disco.
- Es eficiente en SSDs, pero puede afectar la vida útil del almacenamiento si se usa en exceso.

Memoria Comprimida (Compressed Memory)

- En lugar de enviar datos a la memoria virtual inmediatamente, macOS comprime los datos en la RAM para liberar espacio.
- Reduce el uso del disco y mejora el rendimiento del sistema.

Memoria Caché (Disk Cache)

- Guarda datos temporales de aplicaciones y archivos usados recientemente.
- Acelera la apertura de programas y archivos sin recargar datos desde el almacenamiento.

Memoria de GPU (Gráficos)

- Si usas un Mac con una GPU integrada, la memoria RAM se comparte con los gráficos.
- En Macs con GPUs dedicadas, la memoria de video (VRAM) maneja los gráficos de forma independiente.

Memoria NVRAM (Non-Volatile RAM) y PRAM (Parameter RAM)

- Almacena configuraciones básicas como volumen, brillo, fecha/hora y opciones de arranque.
- Es una memoria persistente que no se borra al apagar el equipo.

macOS gestiona la memoria con un enfoque en **eficiencia y optimización automática**, usando tecnologías avanzadas como **Memory Compression** y **Memory Pressure**.

Sistemas de archivos

APFS (Apple File System): Este sistema de archivos fue diseñado por Apple para reemplazar el HFS+. Está optimizado para discos SSD, ofrece soporte para cifrado nativo, snapshots, clonado de archivos y directorios, y es mucho más eficiente en el manejo de archivos.

HFS+ (Hierarchical File System Plus): Fue el sistema de archivos principal de macOS antes de APFS. Soporta journaling y es compatible con discos duros tradicionales (HDD) pero es menos eficiente en discos SSD.

exFAT: Se usa para discos compartidos entre macOS y Windows.

¿Cómo opera en macOS?

- Gestiona la memoria con **swap dinámico** y un sistema de compresión de memoria para optimizar el rendimiento.
- Prefiere APFS por su eficiencia en SSDs.
- Usa técnicas avanzadas como **Memory Pressure**, que redistribuye los recursos entre aplicaciones activas.

Manejo de memoria en diferentes SOs

Sistema Operativo	Gestión de RAM	Memoria Virtual
Windows	Uso de <i>Working Set</i> para cada proceso	Archivo pagefile.sys (ajustable)
Linux	Uso de <i>cgroups</i> y <i>OOM Killer</i>	Partición swap o archivo swapfile
macOS	Optimización con <i>Unified Memory</i>	Uso de <i>dynamic_pager</i> y <i>swapfiles</i>

Resumen Comparativo de sistemas de archivos en los SOs

SO	Sistema de Archivos Nativo	Memoria Virtual	Característica Clave
Windows	NTFS	pagefile.sys	Soporte para EFS y compresión

Linux	ext4	swap (partición/archivo)	Flexibilidad (ext4/XFS/Btrfs)
macOS	APFS	swapfiles dinámicos	Optimizado para SSDs y cifrado



b) Tipos de memorias y sistemas de archivos en dispositivos móviles según 3 sistemas operativos

Android

Tipos de memoria

Maneja la memoria con una combinación de **RAM**, **almacenamiento** y **caché**, optimizando el rendimiento con gestión automática.

RAM: Se usa para ejecutar aplicaciones y procesos en segundo plano. Los dispositivos Android, al igual que las computadoras, usan la RAM para ejecutar aplicaciones activas. La cantidad de RAM es crucial para el rendimiento del dispositivo, permitiendo que más aplicaciones se ejecuten simultáneamente y mejorando la velocidad de respuesta.

Memoria Virtual (Swap/ZRAM): En algunos dispositivos Android, se utiliza una partición o archivo swap para manejar la sobrecarga de RAM, lo que permite mover datos de la RAM a la memoria de almacenamiento cuando esta se agota. Usa una partición en el almacenamiento interno para liberar RAM.

Caché: Almacena datos temporales de aplicaciones y el sistema.

eMMC/UFS (Almacenamiento Interno): Utiliza memorias de tipo **eMMC** o **UFS** según el modelo del dispositivo.

Memoria de GPU (VRAM Compartida): Comparte la RAM con los gráficos.

¿Cómo Opera?

- Android usa **ZRAM** para comprimir datos en la RAM y evitar el uso excesivo del almacenamiento como swap.
- Utiliza **garbage collector (GC)** para limpiar memoria de apps que ya no están en uso.
- Elimina procesos en segundo plano cuando la RAM está saturada.

Sistemas de archivos

Usa **EXT4** en la memoria interna por su velocidad y estabilidad. Es el sistema de archivos más común en dispositivos Android. Ofrece un buen rendimiento en dispositivos de memoria flash y soporta journaling, lo que ayuda a proteger los datos en caso de fallos del sistema.

Tarjetas SD pueden usar **FAT32 o exFAT** (Android 6.0+ permite formatearlas como almacenamiento interno).

F2FS (Flash-Friendly File System): En versiones recientes, **F2FS** se está implementando en algunos dispositivos para mejorar el rendimiento en memorias flash. Fue desarrollado por Samsung y está diseñado específicamente para mejorar el rendimiento en almacenamiento de tipo flash. En algunos dispositivos de gama alta, Android utiliza F2FS debido a su optimización para memorias NAND flash, que son las más comunes en teléfonos y tablets



IOS

Tipos de memoria

iOS tiene una gestión de memoria muy optimizada, con técnicas avanzadas de **compresión y swap en SSD**.

RAM: En los dispositivos iOS (iPhone y iPad), la RAM es clave para el rendimiento. Aunque la cantidad de RAM suele ser menor que en dispositivos Android de gama alta, iOS gestiona la memoria de manera muy eficiente, limitando los procesos en segundo plano y optimizando el uso de recursos para las aplicaciones activas.

Almacenamiento Flash (NVMe): Apple utiliza almacenamiento flash de alta velocidad, comúnmente basado en tecnología NVMe (Non-Volatile Memory Express), que es más rápida que eMMC o UFS. Esto permite tiempos de carga más rápidos y una mejor experiencia de usuario.

Memoria Virtual (Swap): iOS también usa swap, moviendo aplicaciones o procesos que no se están utilizando activamente a la memoria virtual cuando la RAM se llena.

Memoria de GPU (VRAM Compartida): Comparte memoria con la RAM en dispositivos con GPUs integradas.

¿Cómo Opera?

- **No usa swap tradicional;** en su lugar, utiliza **compresión de memoria** para optimizar la RAM.
- Cuando la RAM está llena, **iOS pausa o cierra apps en segundo plano** en lugar de usar almacenamiento como swap.
- Utiliza **pre-fetching** para precargar datos de apps usadas frecuentemente.

Sistemas de archivos

APFS (Apple File System): Introducido en iOS 10.3, APFS está optimizado para dispositivos que usan almacenamiento flash. Es un sistema de archivos avanzado que soporta cifrado de archivos y volúmenes, y una excelente gestión del espacio libre.

iOS se destaca por su gestión eficiente de la memoria, cerrando aplicaciones en segundo plano automáticamente cuando se requiere más espacio en la RAM. APFS asegura un rendimiento óptimo al manejar de manera eficiente la lectura y escritura en almacenamiento flash, además de proteger los datos con cifrado y garantizar la integridad mediante snapshots.

Usa **APFS**, diseñado para **SSDs NVMe** y con soporte de **encriptación nativa**.

No permite acceder directamente al sistema de archivos a menos que el dispositivo esté "jailbreakeado".

Compatible con **exFAT** para discos externos conectados vía USB-C o adaptadores Lightning.



HarmonyOS

Tipos de memoria

Combina **arquitectura distribuida y memoria optimizada** para dispositivos inteligentes.

RAM: Se gestiona con un sistema dinámico de liberación de memoria.

Memoria Virtual (Dynamic Paging): Utiliza almacenamiento interno como memoria virtual cuando es necesario.

Caché: Usa un sistema de caché distribuida para compartir datos entre dispositivos conectados.

UFS/NVMe (Almacenamiento Interno): Huawei usa **UFS 3.1/4.0** o NVMe en dispositivos premium.

Memoria de GPU (VRAM Compartida): Comparte memoria con la GPU integrada para mejorar gráficos.

¿Cómo Opera?

- **Gestión de Memoria Distribuida:** Permite compartir recursos de RAM entre dispositivos del ecosistema HarmonyOS.
- Usa **compresión de memoria y técnicas de IA** para optimizar la RAM.
- Soporta **carga rápida de procesos y multitarea avanzada** con su kernel optimizado.

Sistemas de archivos

HarmonyOS File System: HarmonyOS utiliza un sistema de archivos optimizado para almacenamiento flash, similar a F2FS en Android. Está diseñado para mejorar la eficiencia en la lectura y escritura de datos en dispositivos móviles, especialmente cuando se usan memorias flash NAND.

HarmonyOS opera mediante una arquitectura distribuida que permite la interconexión eficiente entre diferentes dispositivos, compartiendo memoria y almacenamiento cuando sea necesario. Su sistema de archivos está optimizado para mejorar la vida útil de la memoria flash y el rendimiento de los dispositivos móviles.

Usa **ERFS (Enhanced Read-Only File System)**, un sistema propietario de Huawei optimizado para velocidad y seguridad.

Permite acceso rápido a datos con técnicas de **almacenamiento en caché**.

Compatible con **exFAT y EXT4** para almacenamiento externo.



Resumen Comparativo de sistemas de archivos en los SOs para dispositivos móviles

Sistema Operativo	Sistema de Archivos Principal	Compatibilidad	Optimización
Android	EXT4 (para la memoria interna)	FAT32, exFAT (tarjetas SD)	Soporta encriptación y journaling
iOS	APFS (Apple File System)	exFAT (para dispositivos externos)	Optimizado para SSDs NVMe y encriptación avanzada
HarmonyOS	ERFS (Enhanced Read-Only File System)	exFAT, EXT4 (para dispositivos externos)	Usa compresión y carga rápida

Cuadro comparativo de los tipos de memoria y los sistemas de archivo de cada SOs

Sistema Operativo	Tipos de Memoria	Gestión de Memoria	Sistema de Archivos
-------------------	------------------	--------------------	---------------------

Android	RAM, Swap (ZRAM), Caché, eMMC/UFS	ZRAM, Garbage Collector, Eliminación de procesos	EXT4, FAT32/exFAT (externo)
iOS	RAM, Compressed Memory, Caché, NVMe	Memory Compression, Cierre de Apps en segundo plano	APFS, exFAT (externo)
HarmonyOS	RAM, Dynamic Paging, Caché, UFS/NVMe	Memoria Distribuida, AI Memory Optimization	ERFS, exFAT/EXT4 (externo)

c) **Sistemas operativos de red**

Un sistema operativo de red es un software que permite la interconexión de ordenadores para poder acceder a los servicios y recursos, hardware y software creando redes entre ordenadores. Su función principal es permitir la comunicación entre dispositivos, el acceso compartido a archivos y la gestión de seguridad en entornos de red, facilitando el trabajo colaborativo y el uso eficiente de los recursos informáticos.

La composición principal de un hardware que utiliza un sistema operativo de red incluye varios ordenadores, impresora y dos servidores, uno de ellos para archivos con una red local que los conecta entre sí. Su función es proporcionar servicios y características básicas de red que soporten múltiples solicitudes de entrada simultáneamente para entornos multiusuarios.

Los sistemas operativos de red pueden clasificarse de diferentes formas, pero lo más común es clasificarlo según el sistema operativo de base.

Aplicaciones en la Vida Diaria

Los sistemas operativos de red están presentes en diversas áreas de nuestra vida cotidiana. Por ejemplo, en empresas y oficinas, permiten la administración de servidores que almacenan bases de datos y documentos accesibles para múltiples usuarios. En el ámbito educativo, los sistemas operativos de red posibilitan la gestión de plataformas de aprendizaje en línea y el acceso a bibliotecas digitales. Asimismo, en hogares con múltiples dispositivos conectados, estos sistemas ayudan a administrar redes domésticas, permitiendo la comunicación entre computadoras, impresoras y otros dispositivos inteligentes.

Tipos de Sistemas Operativos de Red

Existen diversos tipos de sistemas operativos de red, cada uno diseñado para cumplir funciones específicas según el entorno en el que se implementen:

1. **Sistemas Operativos de Red Peer-to-Peer (P2P):** En este tipo de sistemas, todos los dispositivos tienen igualdad de condiciones y pueden actuar tanto como clientes como servidores. Son comunes en redes domésticas o en pequeñas oficinas donde los recursos se comparten sin la necesidad de un servidor central.
2. **Sistemas Operativos de Red Basados en Servidor:** Se caracterizan por contar con un servidor central que administra los recursos y el acceso de los clientes dentro de la red. Ejemplos de estos sistemas incluyen Windows Server, Linux (con distribuciones como Ubuntu Server y Red Hat Enterprise

Linux) y macOS Server. Son ampliamente utilizados en empresas y organizaciones que requieren mayor seguridad y control en sus redes.

3. **Sistemas Operativos de Red Distribuidos:** En estos sistemas, múltiples computadoras trabajan juntas como si fueran una sola, compartiendo recursos y procesando información de manera coordinada. Son comunes en centros de datos y supercomputadoras utilizadas en investigaciones científicas y aplicaciones industriales.
4. **Sistemas Operativos de Red en la Nube:** Con el auge de la computación en la nube, han surgido sistemas operativos diseñados específicamente para gestionar redes basadas en servicios en línea. Ejemplos de estos incluyen Google Chrome OS y Amazon AWS, que permiten acceder a aplicaciones y datos desde cualquier lugar con conexión a Internet.

Sistemas de archivos

Los sistemas operativos de red utilizan una variedad de sistemas de archivos diseñados para compartir y gestionar datos entre múltiples dispositivos en una red. Algunos de los sistemas de archivos más comunes incluyen:

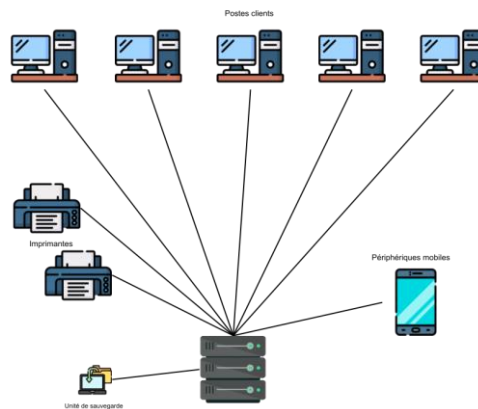
- **NFS (Network File System):** Se utiliza principalmente en sistemas basados en Unix y Linux. Este sistema de archivos permite a los usuarios acceder a archivos y directorios en otros equipos como si estuvieran en el sistema local. NFS es muy utilizado en entornos Linux y ofrece características como la compatibilidad con grandes archivos y mejoras en el rendimiento y la seguridad en su versión 4.
- **SMB (Server Message Block):** Utilizado principalmente en redes de Windows, SMB permite compartir archivos, impresoras y otros recursos. SMB es común en entornos empresariales con servidores Windows, y sus versiones más recientes, como SMBv3, incluyen mejoras significativas en seguridad, como el cifrado de extremo a extremo y la autenticación más robusta.
- **AFP (Apple Filing Protocol):** Utilizado en sistemas operativos macOS, AFP fue desarrollado para compartir archivos en redes Apple. Aunque ha sido en gran medida reemplazado por SMB y NFS en entornos más modernos, aún se utiliza en algunos sistemas más antiguos o especializados.

Aplicaciones y Beneficios

Los sistemas operativos de red tienen una amplia gama de aplicaciones en distintos sectores. En el ámbito empresarial, facilitan la administración centralizada de usuarios y la seguridad de la información. En el sector educativo, posibilitan el acceso a plataformas digitales y la gestión eficiente de redes escolares. Además, en el mundo del entretenimiento y la tecnología, estos sistemas permiten la interconexión de servidores de videojuegos en línea y servicios de streaming, optimizando la experiencia del usuario.

Entre los beneficios más destacados de los sistemas operativos de red se encuentran la eficiencia en la gestión de recursos, la seguridad mejorada mediante autenticación y permisos de acceso, y la escalabilidad, que permite adaptar la infraestructura de red a medida que crece una organización o comunidad.

Los sistemas operativos de red desempeñan un papel fundamental en la administración de redes informáticas, facilitando la comunicación y el acceso a recursos compartidos en diversos ámbitos de la vida diaria. Su evolución continúa con la integración de tecnologías como la computación en la nube y la inteligencia artificial, lo que promete mejorar aún más la eficiencia y seguridad de las redes en el futuro. Conocer estos sistemas es clave para comprender el funcionamiento de la infraestructura digital que sostiene nuestra sociedad moderna.



¿Porque es importante la administración de memoria en estos dispositivos?

La administración de memoria en un sistema operativo es fundamental porque permite que los programas y procesos se ejecuten de manera eficiente y estable. Sin una gestión adecuada, los dispositivos podrían experimentar problemas como ralentización, bloqueos o incluso fallos del sistema.

Uno de los principales objetivos de la administración de memoria es asignar y liberar espacio de manera dinámica, asegurando que cada aplicación reciba la cantidad de memoria que necesita sin desperdiciar recursos. Esto es especialmente importante en sistemas que ejecutan múltiples programas simultáneamente, como computadoras personales, servidores y dispositivos móviles.

Funciones clave de la administración de memoria

1. **Asignación de memoria:** El sistema operativo distribuye la memoria entre los procesos en ejecución y ajusta la asignación según sea necesario.
2. **Paginación y segmentación:** Técnicas utilizadas para dividir la memoria en secciones manejables, optimizando el acceso y evitando fragmentación.
3. **Memoria virtual:** Permite que un sistema utilice espacio en el disco como si fuera memoria RAM, aumentando la capacidad aparente de memoria disponible.
4. **Protección y seguridad:** Evita que un proceso acceda a la memoria de otro, reduciendo errores y vulnerabilidades de seguridad.
5. **Liberación de memoria:** Una vez que un proceso termina, la memoria utilizada se libera para que otros procesos puedan utilizarla.

En dispositivos embebidos y en tiempo real, la administración de memoria es aún más crítica, ya que estos sistemas suelen tener recursos limitados y deben responder en tiempos precisos. En sistemas multiusuario o distribuidos, una buena gestión de memoria garantiza que todos los usuarios y procesos tengan acceso equitativo a los recursos sin afectar el rendimiento general del sistema.

Un sistema operativo es un software fundamental que administra los recursos de hardware y software en un dispositivo, facilitando la comunicación entre el usuario y la computadora. Su propósito principal es gestionar la ejecución de programas, la asignación de memoria, el acceso a dispositivos de almacenamiento y la comunicación con periféricos. Además, optimiza el uso de los recursos disponibles,

garantizando que múltiples procesos puedan ejecutarse de manera eficiente y sin interferencias.

Existen distintos tipos de sistemas operativos, cada uno diseñado para satisfacer necesidades específicas según el entorno en el que operan:

- **Monousuario y Multiusuario:** Un sistema operativo monousuario permite que un solo usuario interactúe con el sistema a la vez, mientras que un sistema multiusuario posibilita que varios usuarios accedan simultáneamente, compartiendo recursos de manera eficiente.
- **Sistemas en Tiempo Real:** Estos sistemas están diseñados para responder en un tiempo predecible a eventos externos, siendo utilizados en entornos críticos como la industria automotriz, la aviación y los dispositivos médicos.
- **Sistemas Distribuidos:** Permiten la gestión y ejecución de procesos en múltiples dispositivos interconectados, optimizando el rendimiento y asegurando redundancia para evitar fallos.
- **Sistemas Embebidos:** Son sistemas operativos diseñados para dispositivos específicos, como electrodomésticos, sistemas de automatización y dispositivos IoT, optimizando el consumo de recursos y asegurando un funcionamiento estable y eficiente.

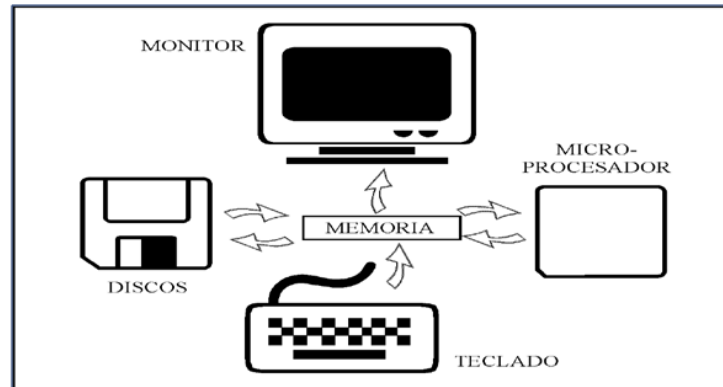
Los sistemas operativos también incluyen interfaces de usuario, que pueden ser de tipo gráfico (GUI) o basadas en línea de comandos (CLI). Las interfaces gráficas permiten a los usuarios interactuar de manera intuitiva a través de ventanas, iconos y menús, mientras que las interfaces en línea de comandos requieren el uso de instrucciones escritas, proporcionando mayor control y flexibilidad a usuarios avanzados.

Además, los sistemas operativos incorporan diversos mecanismos de seguridad para proteger la integridad de los datos y la estabilidad del sistema. Entre estos mecanismos se encuentran el control de acceso mediante credenciales, la encriptación de datos, los sistemas de detección de intrusos y las actualizaciones constantes para corregir vulnerabilidades.

Algunos ejemplos comunes de sistemas operativos incluyen:

- **Windows:** Desarrollado por Microsoft, es ampliamente utilizado en computadoras personales y entornos empresariales.
- **macOS:** Sistema operativo exclusivo de Apple, reconocido por su estabilidad y seguridad.

- **Linux:** Un sistema operativo de código abierto con múltiples distribuciones adaptadas a diversas necesidades.
- **Android:** Basado en Linux, es el sistema operativo más utilizado en dispositivos móviles.
- **iOS:** Sistema operativo desarrollado por Apple para sus dispositivos móviles, optimizado para el ecosistema de la marca.



¿Qué debe hacer el sistema operativo para llevar un control de los espacios disponibles y ocupados en la memoria?

Para llevar un control eficiente de los espacios disponibles y ocupados en la memoria, el sistema operativo utiliza diversas técnicas y estructuras de administración de memoria. Esto es esencial para garantizar un uso óptimo de los recursos y evitar problemas como la fragmentación, la sobresaturación o la corrupción de datos.

Técnicas de Administración de Memoria

1. Tablas de asignación de memoria

- a. El sistema operativo mantiene registros de qué áreas de la memoria están en uso y cuáles están libres. Esto se logra mediante estructuras como tablas de páginas o tablas de segmentos, que permiten asignar y liberar memoria de manera eficiente. Estas tablas contienen información sobre la ubicación y el tamaño de los bloques de memoria, asegurando que cada proceso reciba la cantidad de memoria necesaria sin interferir con otros procesos en ejecución.

2. Mapeo de memoria y gestión de direcciones

- a. Se emplean esquemas como la paginación y la segmentación para dividir la memoria en bloques manejables. La paginación divide la memoria en unidades de tamaño fijo llamadas páginas, mientras que la segmentación la divide según la lógica del programa, facilitando el acceso y la organización de los datos. Esto permite que los procesos se ejecuten sin necesidad de ocupar un espacio contiguo en la memoria, optimizando su uso.

3. Memoria virtual

- a. Para evitar la saturación de la memoria física, el sistema operativo puede utilizar parte del almacenamiento secundario (disco duro o SSD) como una extensión de la memoria RAM. Este mecanismo permite manejar más procesos de los que cabrían en la memoria principal, aunque con un menor rendimiento debido a la diferencia en velocidad entre la RAM y el disco de almacenamiento. La memoria virtual se implementa mediante un proceso conocido como **swapping**, en el que las páginas de menor prioridad se trasladan temporalmente al disco para liberar espacio en la memoria RAM.

4. Gestión de fragmentación

- a. Con el tiempo, la memoria puede fragmentarse debido a la asignación y liberación de espacios de diferentes tamaños. La fragmentación puede ser interna, cuando hay espacio desperdiciado dentro de un

bloque asignado, o externa, cuando hay suficiente memoria libre, pero en bloques dispersos que no pueden ser utilizados de manera eficiente. Para contrarrestar esto, el sistema operativo emplea estrategias como la **compactación**, que reorganiza los bloques de memoria en uso para consolidar el espacio libre, y el uso de listas enlazadas, que facilitan la gestión de memoria en bloques no contiguos.

5. Monitoreo y asignación dinámica

- a. Los sistemas operativos modernos implementan algoritmos de asignación dinámica, como:
 - i. **First-Fit**: Asigna el primer bloque de memoria lo suficientemente grande para el proceso.
 - ii. **Best-Fit**: Selecciona el bloque de memoria más pequeño posible que sea adecuado para el proceso, minimizando la fragmentación.
 - iii. **Worst-Fit**: Asigna el bloque de memoria más grande disponible, dejando fragmentos grandes que pueden ser reutilizados más fácilmente.

Estos algoritmos permiten mejorar el rendimiento y reducir el desperdicio de memoria al elegir la mejor estrategia para cada situación.

6. Protección y seguridad en la memoria

- a. Para evitar que un proceso acceda a la memoria asignada a otro, el sistema operativo implementa mecanismos de protección. Entre estos mecanismos se encuentran el uso de **espacios de direcciones separadas**, donde cada proceso tiene su propio espacio de memoria, y la implementación de permisos de acceso, que determinan qué procesos pueden leer, escribir o ejecutar en ciertas áreas de memoria. Además, se utilizan técnicas de encriptación y aislamiento para proteger datos sensibles contra accesos no autorizados y ataques cibernéticos.

7. Manejo de errores y recuperación

- a. Durante la ejecución de programas, pueden ocurrir errores de acceso a memoria, como intentos de escritura en áreas protegidas o acceso a direcciones inexistentes. El sistema operativo detecta estos errores y responde con mecanismos de recuperación, como la terminación del proceso afectado o la asignación de un nuevo bloque de memoria. También implementa registros de auditoría para analizar fallos y mejorar la estabilidad del sistema.

Gracias a estas técnicas, el sistema operativo puede administrar eficientemente la memoria, asegurando que los programas tengan suficiente espacio para ejecutarse sin desperdiciar recursos ni generar conflictos entre procesos. La gestión eficiente de la memoria no solo optimiza el rendimiento del sistema, sino que también mejora la seguridad y la estabilidad de las aplicaciones en ejecución



¿Qué problemas son más frecuentes en la administración de memoria?

La administración de memoria es un aspecto fundamental en el diseño y funcionamiento de los sistemas operativos. Su correcta gestión permite un uso eficiente de los recursos, garantizando la estabilidad y el rendimiento de los dispositivos. Sin embargo, diversos problemas pueden afectar este proceso, generando ralentizaciones, errores y fallos en el sistema. Este ensayo analiza los problemas más comunes en la administración de memoria en sistemas operativos como Windows, Linux, macOS, Android, iOS y HarmonyOS, así como sus soluciones y estrategias de mitigación.

Problemas Comunes en la Administración de Memoria

Fragmentación de Memoria

La fragmentación ocurre cuando la memoria se divide en pequeños bloques dispersos, lo que impide la asignación contigua de grandes espacios de memoria. Existen dos tipos principales:

- **Fragmentación interna:** Ocurre cuando se asigna un bloque de memoria mayor al requerido, dejando espacios desperdiciados dentro de los bloques.
- **Fragmentación externa:** Se produce cuando hay suficiente memoria libre, pero no en bloques contiguos suficientemente grandes para satisfacer una solicitud.

Casos específicos:

- En Windows, la fragmentación externa puede afectar el rendimiento de la memoria virtual.
- En Linux, los sistemas de archivos pueden sufrir fragmentación si no se optimizan adecuadamente.
- En dispositivos Android e iOS, la memoria flash puede fragmentarse, reduciendo la vida útil del almacenamiento.

Soluciones:

- Uso de **compactación de memoria** para reorganizar los bloques.
- Implementación de **asignación dinámica de memoria** con mejores algoritmos de gestión.
- En discos duros, herramientas como **desfragmentadores** ayudan a reorganizar los datos (Windows).

Fugas de Memoria

Las fugas de memoria ocurren cuando un programa no libera correctamente la memoria asignada, dejando fragmentos inútiles que el sistema no puede reutilizar.

Casos específicos:

- En aplicaciones de Windows, programas mal optimizados pueden acumular memoria inutilizable.
- En Linux y macOS, errores en la programación de procesos pueden generar leaks en memoria RAM.
- En Android e iOS, algunas apps pueden mantener procesos en segundo plano, consumiendo memoria de forma innecesaria.

Soluciones:

- Uso de **recolectores de basura** en lenguajes como Java y Python.
- Herramientas como **Valgrind** en Linux o **Instruments** en macOS/iOS para detectar fugas.
- Restricción del uso de apps en segundo plano en sistemas móviles.

Uso Excesivo de Memoria Virtual

Cuando la RAM se llena, el sistema usa el almacenamiento como memoria virtual, lo que puede reducir el rendimiento.

Casos específicos:

- En Windows, el uso excesivo del archivo de paginación puede causar ralentizaciones en HDDs.
- En Linux, el uso agresivo del swap puede hacer que el sistema se vuelva lento.
- En macOS, aunque el swap está optimizado, un uso excesivo puede desgastar los SSDs.

Soluciones:

- Incrementar la RAM para reducir la dependencia de la memoria virtual.
- Ajustar la configuración de swap según el uso del sistema.
- En Linux, modificar el parámetro **swappiness** para evitar que el swap se use antes de lo necesario.

Condiciones de Carrera y Deadlocks

Cuando varios procesos intentan acceder a la misma memoria sin un control adecuado, pueden generarse bloqueos o ejecuciones indefinidas.

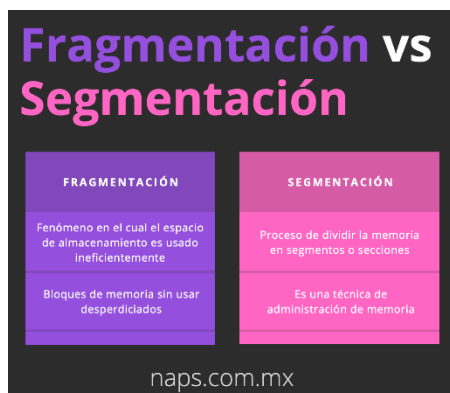
Casos específicos:

- En sistemas multiusuario de Linux, procesos pueden quedar bloqueados por mala gestión de semáforos.
- En Windows, aplicaciones que comparten recursos pueden generar interbloqueos.
- En sistemas móviles, juegos y aplicaciones pesadas pueden colapsar por mal manejo de memoria compartida.

Soluciones:

- Uso de **mutex y semáforos** para regular el acceso a la memoria compartida.
- Implementación de **algoritmos de detección y recuperación de deadlocks**.
- En sistemas críticos, estrategias de **prevención de interbloqueos** como el protocolo del banquero.

La administración de memoria es crucial para el rendimiento y estabilidad de los sistemas operativos. Problemas como la fragmentación, fugas de memoria, uso excesivo de memoria virtual y condiciones de carrera pueden afectar negativamente el funcionamiento del sistema. Sin embargo, existen soluciones efectivas, como el uso de recolectores de basura, optimización de swap y el manejo adecuado de procesos concurrentes. La mejora continua en la gestión de memoria es clave para garantizar sistemas más eficientes y confiables en el futuro



Conclusión

A través de la gestión de la memoria y la organización de archivos, cada sistema operativo ofrece una experiencia única que afecta la eficiencia, la seguridad y la accesibilidad de la información. Desde la rapidez con la que podemos acceder a nuestros datos hasta las estrategias implementadas para proteger nuestra privacidad, cada detalle cuenta en la manera en que utilizamos nuestros dispositivos.

Al comprender las diferencias y similitudes entre los sistemas de almacenamiento, podemos tomar decisiones más informadas que optimicen nuestra productividad y seguridad. A medida que la tecnología continúa evolucionando y la cantidad de datos que generamos sigue creciendo, se vuelve esencial apreciar la complejidad de estos procesos y cómo influyen en nuestra vida cotidiana.

La administración de memoria es un factor determinante en la eficiencia y estabilidad de los sistemas operativos. Problemas como la fragmentación, las fugas de memoria y el uso excesivo de memoria virtual pueden degradar el rendimiento del sistema y generar errores críticos. Sin embargo, las soluciones actuales, como la compactación de memoria, el uso de algoritmos de optimización y la gestión inteligente de caché, han permitido mejorar significativamente la eficiencia del uso de la memoria en diversos sistemas operativos. Con el avance de la tecnología y la creciente demanda de dispositivos con mayor capacidad de procesamiento, la gestión de memoria seguirá evolucionando para garantizar un mejor aprovechamiento de los recursos y una experiencia de usuario óptima.

Conclusiones individuales

Jose Ángel Cardenas Contreras, 1935156

En conclusión, los componentes más fundamentales en un sistema para su correcto funcionamiento son el almacenaje, la memoria y los archivos, debido a que cada uno permite el funcionamiento y complementa los demás componentes por ejemplo la RAM permite un acceso rápido a los datos mientras se utiliza el sistema mientras que el almacenaje como el HDD o SSD, guarda la información fija. Los archivos organizan toda la información y facilitan su gestión dentro del sistema

Frida Jaziry Juarez Fuentes, 2028420

El almacenaje, la memoria y los archivos son componentes fundamentales en la gestión y funcionamiento de los sistemas informáticos. La memoria, ya sea volátil (RAM) o no volátil (ROM), permite el acceso rápido a los datos y procesos en ejecución, mientras que el almacenamiento a largo plazo (como discos duros y unidades SSD) garantiza la conservación de la información incluso cuando el sistema está apagado. Los archivos organizan y estructuran los datos, facilitando su acceso y manipulación. Una gestión eficiente de estos elementos es clave para el rendimiento, la seguridad y la integridad de la información en cualquier entorno digital.

Fátima Arizpe Sánchez, 2025106

El almacenaje, la memoria y los archivos son fundamentales para el funcionamiento de cualquier sistema informático. La memoria RAM permite un acceso rápido a los datos mientras el sistema está en uso, mientras que el almacenamiento permanente (HDD, SSD) guarda la información a largo plazo. Los archivos organizan los datos y facilitan su gestión dentro del sistema. En conjunto, estos elementos determinan la velocidad, capacidad y eficiencia del equipo.

Francisco Gael Reyes Cantú, 1995983

La memoria, el almacenamiento y los archivos conforman la base sobre la que operan los sistemas computacionales. La memoria se encarga de guardar temporalmente los datos que se utilizan en tiempo real, acelerando la ejecución de procesos y mejorando la respuesta del sistema. El almacenamiento, en cambio, permite conservar información de manera duradera, asegurando su disponibilidad incluso después de apagar el equipo. Los archivos funcionan como estructuras organizadas que permiten clasificar y acceder a la información de forma eficiente. Una administración adecuada de estos recursos es clave para mantener un

rendimiento fluido, proteger los datos y asegurar el correcto funcionamiento en cualquier entorno digital.

Isac Alfredo Almaguer Espinosa, 2049903

Entender la administración de memoria en los sistemas operativos es clave para desarrollar software eficiente y optimizar el rendimiento del hardware. A lo largo de este trabajo, hemos analizado los diferentes tipos de memoria utilizados en sistemas como Windows, Linux, macOS, Android, iOS y HarmonyOS, así como sus sistemas de archivos y la forma en que manejan el almacenamiento. También hemos identificado los principales problemas que afectan la gestión de memoria, como la fragmentación, las fugas de memoria y el uso excesivo de la memoria virtual, los cuales pueden degradar el rendimiento del sistema y generar errores críticos. La importancia que tienen los tipos de memoria y los sistemas de archivos muchas veces no nos damos cuenta, pero están ahí, el haber analizado cada sistema operativo me doy cuenta que, aunque parece que usan los mismos tipos de sistemas son muy diferentes cuando indagas más a profundidad.

Bibliografía

Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). *Operating System Concepts (10th ed.)*. Wiley.

Tanenbaum, A. S., & Bos, H. (2015). *Modern Operating Systems (4th ed.)*. Pearson.

Microsoft. (2023). *How NTFS works*. Microsoft Docs.
<https://learn.microsoft.com/en-us/windows-server/storage/file-server/ntfs-overview>

Russinovich, M. E., Solomon, D. A., & Ionescu, A. (2012). *Windows Internals (6th ed.)*. Microsoft Press.

Love, R. (2010). *Linux Kernel Development (3rd ed.)*. Addison-Wesley Professional.

The Linux Kernel Archives. (2023). *Filesystems in the Linux kernel*.
<https://www.kernel.org/doc/html/latest/filesystems/>

Apple Inc. (2023). *About Apple File System (APFS)*. Apple Developer Documentation.
https://developer.apple.com/documentation/foundation/file_system/about_apple_file_system

Singh, A. (2006). *Mac OS X Internals: A Systems Approach*. Addison-Wesley Professional.

Google. (2023). *File-based encryption*. Android Open Source Project.
<https://source.android.com/docs/security/features/encryption/file-based>

Apple Inc. (2023). *About Apple File System (APFS)*. Apple Developer Documentation.
https://developer.apple.com/documentation/foundation/file_system/about_apple_file_system

Huawei. (2023). *HarmonyOS Technical Whitepaper*. Huawei Developers.
<https://developer.harmonyos.com/en/docs/documentation/doc-guides/harmonyos-overview-00000000000011903>