

Instalación de software libre y propietario (Implantación de sistemas operativos)

Estructura de un sistema informático

La estructura interna de un sistema operativo es transparente al usuario, dependiendo del sistema operativo se estructura de una forma u otra, en este apartado se ven algunas estructuras que se han utilizado y se utilizan en los sistemas operativos:

Sistemas monolíticos

El sistema operativo está formado por un conjunto de procedimientos de forma que cada uno puede llamar a los demás cuando lo necesite. Todas las funciones que realiza un sistema operativo se llevan a cabo con un sólo programa que se ejecuta en modo kernel.

Ejemplos

- <https://www.jesusninoc.com/09/09/comandos-informacion-sobre-el-kernel-de-linux/>
 - <https://www.jesusninoc.com/09/09/obtener-informacion-sobre-la-version-de-windows-con-powershell/>
-

Ejercicios

- <https://www.jesusninoc.com/09/14/guardar-en-un-fichero-la-version-de-windows-utilizando-un-cmdlet/>
 - <https://www.jesusninoc.com/09/14/guardar-en-un-fichero-la-version-del-nucleo-de-linux-wsl-utilizando-un-cmdlet-de-powershell/>
-

Sistemas de capas

Consiste en organizar el sistema operativo mediante una jerarquía de capas, cada capa tiene una función.

Microkernels

La idea de esta estructura es dividir el sistema operativo en módulos pequeños y sólo uno se ejecuta en modo kernel, de esta forma un error en un módulo no afecta a todo el sistema, además tienen la ventaja de que son fáciles de mantener debido a la estructura modular que poseen.

Máquinas virtuales

Se trata de realizar copias exactas de hardware que tiene la máquina real incluyendo modo kernel y modo usuario, la entrada y salida, etc. Cada máquina virtual es idéntica al verdadero hardware y cada una puede ejecutar un sistema operativo distinto.

Exokernels

Consiste en dividir los recursos y asignar a cada usuario una parte de esos recursos.

Arquitectura de un sistema operativo

La organización de los sistemas operativos ha evolucionado pero los siguientes componentes deben estar incluidos en todo sistema operativo:

Ejercicio

- <https://www.jesusninoc.com/05/05/conocer-la-arquitectura-del-sistema-operativo-con-powershell/>
-

Núcleo

El núcleo es el componente del sistema operativo que siempre está residente en la memoria real del ordenador. La función principal del núcleo es transformar los recursos reales del ordenador en recursos estándares y cómodos de usar.

Más información

- <https://www.jesusninoc.com/11/01/ver-informacion-sobre-el-nucleo-de-linux/>
-

Drivers

Los dispositivos de entrada y salida utilizados para comunicarse con el ordenador, necesitan dos elementos: un dispositivo controlador y el dispositivo en sí. El dispositivo controlador es un chip o un conjunto de chips que controlan físicamente el dispositivo. La comunicación entre el dispositivo controlador y el sistema operativo se realiza mediante un software llamado driver.

Más información

- <https://www.jesusninoc.com/02/12/obtener-informacion-sobre-los-controladores-usb/>
-

El sistema de archivos

El sistema de archivos se puede definir como un conjunto de normas y procedimientos para almacenar información en los dispositivos de almacenamiento.

Más información

- <https://www.jesusninoc.com/07/04/4-gestion-del-sistema-de-archivos-en-powershell/>
-

El intérprete de comandos

El intérprete de comando o shell se encarga de leer las órdenes interactivas del usuario y ejecutar los programas que el usuario indique.

Más información

- https://www.jesusninoc.com/07/03/3-gestion-del-hardware-en-powershell/#Dispositivos_de_entrada_y_salida
-

Ejercicios

- <https://www.jesusninoc.com/09/29/ejemplos-de-ejecucion-de-comandos-en-bash-y-powershell-parte-1/>
 - <https://www.jesusninoc.com/09/30/ejemplos-de-ejecucion-de-comandos-en-bash-y-powershell-parte-2/>
 - <https://www.jesusninoc.com/10/01/ejemplos-de-ejecucion-de-comandos-en-bash-y-powershell-parte-3/>
-

Funciones de un sistema operativo

Funciones principales que realiza el sistema operativo:

Gestión de procesos

El elemento principal es el proceso que se define como programa en ejecución. El sistema operativo se encarga de: crear y destruir procesos; suspender y reanudar procesos;

sincronizar y comunicar procesos.

Ejemplos

Arrancar procesos

Parar procesos

Comparar procesos

- https://www.jesusninoc.com/07/01/1-introduccion-a-powershell/#Ver_los_nombres_procesos_que_se_estan_ejecutando_en_dos_momentos_distintos_y_comprar_el_resultado

Procesos y ruta de ejecución del proceso

- <https://www.jesusninoc.com/11/29/ejercicios-de-powershell-procesos-y-ruta-de-ejecucion-del-proceso/>

Analizar servicios

- <https://www.jesusninoc.com/11/10/analizar-servicios-con-powershell/>

Información sobre el procesador

- <https://www.jesusninoc.com/12/22/ejercicios-de-powershell>

[l-obtener-informacion-sobre-el-procesador/](#)

Información sobre la arquitectura

- <https://www.jesusninoc.com/12/01/ejercicios-de-powershell-obtener-informacion-sobre-la-arquitectura/>

Información sobre la organización

- <https://www.jesusninoc.com/12/02/ejercicios-de-powershell-obtener-informacion-sobre-la-organizacion/>

Información sobre la realización

- <https://www.jesusninoc.com/12/07/ejercicios-de-powershell-obtener-informacion-sobre-la-realizacion/>

Información sobre los procesadores

- <https://www.jesusninoc.com/10/13/mostrar-informacion-sobre-los-procesadores/>

Número de cores

- <https://www.jesusninoc.com/10/02/ejercicios-de-powershell-informacion-sobre-el-numero-de-cores/>

Carga del procesador

- <https://www.jesusninoc.com/10/03/ejercicios-de-powershell-ver-la-carga-del-procesador/>

Información sobre el procesador

- <https://www.jesusninoc.com/01/13/ver-informacion-del-pro>

Ejercicios

- https://www.jesusninoc.com/09/14/ejercicios-de-powershell-almacenar-en-un-fichero-los-procesos-que-se-estan-ejecutando-y-la-fecha-actual/#Solucion_simple
- https://www.jesusninoc.com/09/14/ejercicios-de-powershell-almacenar-en-un-fichero-los-procesos-que-se-estan-ejecutando-y-la-fecha-actual/#Solucion_avanzada_creando_alias

Juntar dos ficheros con información (fichero con fecha y fichero con procesos que se ejecutan):

- <https://www.jesusninoc.com/11/02/ejercicios-de-powershell-juntar-dos-ficheros-con-informacion-fichero-con-fecha-y-fichero-con-procesos-que-se-ejecutan/>

Ejecutar un comando que se encuentra escrito en una cadena:

- <https://www.jesusninoc.com/11/03/ejercicios-de-powershell-ejecutar-un-comando-que-se-encuentra-escrito-en-una-cadena/>

Mostrar los procesos con un Where:

- <https://www.jesusninoc.com/11/04/ejercicios-de-powershell-mostrar-informacion-sobre-un-proceso-que-se-esta-ejecutando-con-un-where/>

Abrir procesos leyendo desde un fichero (en un fichero aparece

el número de veces que se abre el proceso y en otro fichero aparece el nombre del proceso que queremos abrir)

- <https://www.jesusninoc.com/11/05/ejercicios-de-powershell-abrir-procesos-leyendo-desde-un-fichero-en-un-fichero-aparece-el-numero-de-veces-que-se-abre-el-proceso-y-en-otro-fichero-aparece-el-nombre-del-proceso-que-queremos-abrir/>

Parar el proceso leyendo desde un fichero el nombre del proceso que vamos a parar (el nombre del proceso que vamos a parar se almacena mediante un Out-File siendo el primer proceso)

- <https://www.jesusninoc.com/11/06/ejercicios-de-powershell-parar-el-proceso-leyendo-desde-un-fichero-el-nombre-del-proceso-que-vamos-a-parar-el-nombre-del-proceso-que-vamos-a-parar-se-almacena-mediante-un-out-file-siendo-el-primer-p/>

Ejercicios básicos sobre procesos en PowerShell (con alias)

- <https://www.jesusninoc.com/09/22/ejercicios-de-powershell-ejercicios-basicos-sobre-procesos-en-powershell-con-alias/>

Ejercicios básicos sobre procesos en Linux

- <https://www.jesusninoc.com/03/11/ejercicios-basicos-sobre-procesos-en-linux/>

Ejercicios de repaso sobre procesos

- <https://www.jesusninoc.com/09/23/ejercicios-de-powershell-mostrar-los-procesos-ordenados-por-cpu-cuyo-nombre-indica-el-usuario-utilizar-un-fichero-para-almacenar->

[dicho-nombre/](#)

- <https://www.jesusninoc.com/09/23/ejercicios-de-powershell-mostrar-los-procesos-cuyo-nombre-y-tiempo-de-consumo-de-cpu-indica-el-usuario-y-ordenarlos-por-cpu-utilizar-dos-ficheros-para-almacenar-el-nombre-y-el-tiempo-de-consumo/>
- <https://www.jesusninoc.com/09/23/ejercicios-de-powershell-mostrar-un-numero-de-procesos-por-nombre-que-indica-el-usuario-y-ordenarlos-por-cpu/>
- <https://www.jesusninoc.com/01/01/ejercicios-de-powershell-ejercicios-de-hilos/>

Obtener el nombre de un proceso y el número de hilos de cada proceso

- <https://www.jesusninoc.com/09/28/ejercicios-de-powershell-obtener-el-nombre-de-un-proceso-y-el-numero-de-hilos-de-cada-proceso/>

Obtener el identificador del hilo junto con el nombre del proceso que lo creó

- <https://www.jesusninoc.com/09/28/ejercicios-de-powershell-obtener-el-identificador-del-hilo-junto-con-el-nombre-del-proceso-que-lo-creo/>

Ejercicios de PowerShell: conocer los procesos padres y abuelos (padres de padres) de todos los hilos que se están ejecutando en el sistema

- <https://www.jesusninoc.com/09/28/ejercicios-de-powershell-conocer-los-procesos-padres-y-abuelos-padres-de-padres-de-todos-los-hilos-que-se-estan-ejecutando-en-el-sistema/>

Ejercicios de PowerShell: pedir al usuario un número de hilo y mostrar el nombre del proceso padre que crea el hilo y el abuelo

- <https://www.jesusninoc.com/09/28/ejercicios-de-powershell-pedir-al-usuario-un-numero-de-hilo-y-mostrar-el-nombre-del-proceso-padre-que-crea-el-hilo-y-el-abuelo/>

Almacenar información sobre los hilos

- <https://www.jesusninoc.com/11/22/ejercicios-de-powershell-almacenar-informacion-sobre-hilos-de-forma-continuada/>

Mostrar información sobre los hilos del proceso Notepad

- <https://www.jesusninoc.com/11/17/ejercicios-de-powershell-mostrar-informacion-sobre-los-hilos-del-proceso-notepad/>

Mostrar información sobre los hilos de todos los procesos que se están ejecutando (qué proceso ejecuta el hilo)

- <https://www.jesusninoc.com/11/19/ejercicios-de-powershell-mostrar-informacion-sobre-los-hilos-de-todos-los-procesos-que-se-estan-ejecutando-que-proceso-ejecuta-el-hilo/>

Listar hilos de cada proceso

- <https://www.jesusninoc.com/07/22/listar-los-hilos-de-un-proceso/>

Almacenar información sobre hilos de forma continuada

- <https://www.jesusninoc.com/11/22/ejercicios-de-powershell-almacenar-informacion-sobre-hilos-de-forma-continuada/>

Mostrar los hilos de cada proceso utilizando WMI y el cmdlet Get-Process

- <https://www.jesusninoc.com/11/23/ejercicios-de-powershell-mostrar-los-hilos-de-cada-proceso-utilizando-wmi-y-el-cmdlet-get-process/>

Leer nombre de proceso de un fichero y mostrar hilos (bash en Linux)

- <https://www.jesusninoc.com/03/11/ejercicios-de-bash-leer-nombre-de-proceso-de-un-fichero-y-mostrar-hilos/>

Mostrar el nombre de los procesos leyendo de un fichero el identificador de proceso (bash en Linux)

- <https://www.jesusninoc.com/03/11/ejercicios-de-bash-mostrar-el-nombre-de-los-procesos-leyendo-de-un-fichero-el-identificador-de-proceso/>

Sacar los hilos de los procesos leyendo el nombre del proceso de un fichero (bash en Linux)

- <https://www.jesusninoc.com/10/29/ejercicios-de-bash-sacar-los-hilos-de-los-procesos-leyendo-el-nombre-del-proceso-de-un-fichero/>

Relación entre servicios y procesos

- <https://www.jesusninoc.com/10/11/mostrar-la-relacion-entre-servicios-y-procesos-con-powershell/>

Leer nombre de proceso de un fichero y mostrar hilos

- <https://www.jesusninoc.com/11/21/ejercicios-de-powershell-leer-nombres-de-procesos-de-un-fichero-y-mostrar->

[hilos-de-cada-proceso/](#)

Mostrar el nombre de los procesos leyendo de un fichero el identificador de proceso

- <https://www.jesusninoc.com/03/11/ejercicios-de-powershell-mostrar-el-nombre-de-los-procesos-leyendo-de-un-fichero-el-identificador-de-proceso/>

Leer identificadores de procesos y comprobar que tienen un proceso

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-leer-identificadores-de-procesos-y-comprobar-que-tienen-un-proceso/>

Detectar si hay un tipo de proceso y producto de software que se llame «Acrobat»

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-detectar-si-hay-un-tipo-de-proceso-y-producto-de-software-que-se-llame-acrobat/>

Detectar que se está consumiendo tráfico de red por parte de un proceso

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-detectar-que-se-esta-consumiendo-trafico-de-red-por-parte-de-un-proceso/>

Sacar los hilos de los procesos leyendo el nombre del proceso de un fichero

- <https://www.jesusninoc.com/11/23/ejercicios-de-powershell-listar-los-hilos-de-los-procesos-leyendo-el-nombre-del-proceso-de-un-fichero/>

Analizar hilos con PowerShell

- <https://www.jesusninoc.com/02/06/analizar-hilos-con-powershell/>

Mostrar el proceso que creó el hilo

- <https://www.jesusninoc.com/03/11/ejercicios-de-powershell-mostrar-el-proceso-que-creo-el-hilo/>

Almacenar el número de procesos e hilos en un fichero junto con la fecha (el script se ejecuta cada cinco segundos)

- <https://www.jesusninoc.com/11/26/ejercicios-de-powershell-almacenar-el-numero-de-procesos-e-hilos-en-un-fichero-junto-con-la-fecha-el-script-se-ejecuta-cada-cinco-segundos/>

Ejercicios de PowerShell: saber la ruta de los servicios que se están ejecutando

- <https://www.jesusninoc.com/10/01/ejercicios-de-powershell-saber-la-ruta-de-los-servicios-que-se-estan-ejecutando/>

Ejercicios de PowerShell: de la lista de hilos que se están ejecutando indicar si alguno es un hilo de un proceso de un servicio

- <https://www.jesusninoc.com/10/01/ejercicios-de-powershell-de-la-lista-de-hilos-que-se-estan-ejecutando-indicar-si-alguno-es-un-hilo-de-un-proceso-de-un-servicio/>

Ejercicios de PowerShell: de la lista de procesos que se están ejecutando indicar si alguno es un proceso de un servicio

- <https://www.jesusninoc.com/10/01/ejercicios-de-powershell-de-la-lista-de-procesos-que-se-estan-ejecutando-indicar-si-alguno-es-un-proceso-de-un-servicio/>

Ejercicios de PowerShell: de los servicios que se están ejecutando indicar el nombre del proceso

- <https://www.jesusninoc.com/10/01/ejercicios-de-powershell-de-los-servicios-que-se-estan-ejecutando-indicar-el-nombre-del-proceso/>

Ejercicios de PowerShell: mostrar el padre de un hilo

- <https://www.jesusninoc.com/09/29/ejercicios-de-powershell-mostrar-el-padre-de-un-hilo/>

¿Procesador es Intel o no?

- <https://www.jesusninoc.com/10/16/ejercicios-de-powershell-saber-si-un-procesador-es-intel-o-no/>

Más información

- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-en-powershell/#Informacion_sobre_procesos
- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-en-powershell/#Tabla_de_procesos
- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-en-powershell/#Estados_de_un_proceso
- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-en-powershell/#Crear_procesos
- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-en-powershell/#Parar_procesos

- https://www.jesusninoc.com/07/07/7-gestion-de-procesos-e-n-powershell/#El_procesador
 - <https://www.jesusninoc.com/07/07/7-gestion-de-procesos-e-n-powershell/#Hilos>
 - https://www.jesusninoc.com/07/07/7-gestion-de-procesos-e-n-powershell/#Comunicacion_entre_procesos
 - https://www.jesusninoc.com/07/07/7-gestion-de-procesos-e-n-powershell/#Segundo_plano
 - <https://www.jesusninoc.com/07/07/7-gestion-de-procesos-e-n-powershell/#Servicios>
-

Gestión de memoria

La memoria principal se encarga de almacenar procesos e información de procesos que se están ejecutando en el procesador. El sistema operativo se encarga de asignar y liberar la memoria; decidir cuanta memoria se asigna a un proceso; controla las partes de la memoria que se están utilizando.

Ejemplos

Utilizar zonas de memoria compartida en Linux mediante WSL desde PowerShell

- <https://www.jesusninoc.com/06/25/utilizar-zonas-de-memoria-compartida-en-linux-mediante-wsl-desde-powershell/>

Uso de las funciones `ftok()`, `shmget()`, `shmat()` y `shmctl()` (ejemplo)

- <https://www.jesusninoc.com/11/22/uso-de-las-funciones-ftok-shmget-shmat-y-shmctl-ejemplo/>

Uso de las funciones `ftok()`, `shmget()`, `shmat()` y `shmctl()`

- <https://www.jesusninoc.com/2014/11/22/uso-de-las-funciones-ftok-shmget-shmat-y-shmctl/>

Código para crear la zona de memoria y almacenar el valor, por ejemplo introducir 5

- <https://www.jesusninoc.com/06/25/utilizar-zonas-de-memoria-compartida-en-linux-mediante-wsl-desde-powershell/>

Código para leer la zona de memoria (se listan los procesos, si se ha introducido un 5, se listan 5 procesos)

- <https://www.jesusninoc.com/06/25/utilizar-zonas-de-memoria-compartida-en-linux-mediante-wsl-desde-powershell/>

Ejercicios

Modificar el contenido de un fichero (leer un número de un fichero y sumar un número, después almacenar el valor)

- <https://www.jesusninoc.com/11/13/ejercicios-de-powershell-modificar-el-contenido-de-un-fichero-leer-un-numero-de-un-fichero-y-sumar-un-numero-despues-almacenar-el-valor/>

Modificar el contenido de un fichero (leer un número de un fichero y restar un número, después almacenar el valor)

- <https://www.jesusninoc.com/11/15/ejercicios-de-powershell-modificar-el-contenido-de-un-fichero-utilizando-variables-leer-un-numero-de-un-fichero-y-restar-un-numero-despues-almacenar-el-valor/>

Modificar el contenido de un fichero utilizando variables (leer un número de un fichero y restar un número, después almacenar el valor)

- <https://www.jesusninoc.com/11/15/ejercicios-de-powershell-modificar-el-contenido-de-un-fichero-utilizando-variables-leer-un-numero-de-un-fichero-y-restar-un-numero-despues-almacenar-el-valor/>
-

Más información

- <https://www.jesusninoc.com/07/03/3-gestion-del-hardware-en-powershell/#Memoria>
-

Gestión de archivos

El archivo se define como conjunto de datos almacenados en un dispositivo de almacenamiento. El sistema operativo gestiona los archivos mediante el sistema de archivos que se define como conjunto de normas y procedimientos para almacenar información en los dispositivos de almacenamiento.

Ejemplos

Ejemplos básicos sobre el sistema de archivos en PowerShell

- <https://www.jesusninoc.com/11/03/ejercicios-de-powershell-ejercicios-basicos-sobre-el-sistema-de-archivos-en-powershell/>

Ejercicios

Ejercicios sobre sistemas de archivos (mezclado con procesos)

- <https://www.jesusninoc.com/11/04/ejercicios-de-powershell-ejercicios-sobre-sistemas-de-archivos-mezclado-con-procesos/>

Crear una carpeta cuyo nombre sea la fecha de hoy y meter dentro de la carpeta un fichero con información sobre el proceso Notepad

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-crear-una-carpeta-cuyo-nombre-sea-la-fecha-de-hoy-y-meter-dentro-de-la-carpeta-un-fichero-con-informacion-sobre-el-proceso-notepad/>

Crear una carpeta cuyo nombre sea la fecha de hoy

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-crear-una-carpeta-cuyo-nombre-sea-la-fecha-de-hoy/>

Crear un fichero con información sobre el proceso Notepad

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-crear-un-fichero-con-informacion-sobre-el-proceso-notepad/>

Mostrar los hilos que tiene Notepad y almacenar los números de identificador de hilo en un fichero

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-mostrar-los-hilos-que-tiene-notepad-y-almacenar-los-numeros-de-identificador-de-hilo-en-un-fichero/>

Mostrar los hilos que tiene Notepad

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-mostrar-los-hilos-que-tiene-notepad/>

Abrir 5 veces Notepad y almacenar información sobre el proceso Notepad en un fichero

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-abrir-5-veces-notepad-y-almacenar-informacion-sobre-el-proceso-notepad-en-un-fichero/>

Almacenar información sobre el proceso Notepad en un fichero

- <https://www.jesusninoc.com/10/28/ejercicios-de-powershell-crear-un-fichero-con-informacion-sobre-el-proceso-notepad/>

Crear dos ficheros TXT, convertirlos a PDF y juntarlos con Poppler

- <https://www.jesusninoc.com/11/26/ejercicios-de-powershell-crear-dos-ficheros-txt-convertirlos-a-pdf-y-juntarlos-con-poppler-utilizando-wsl/>

Crear cinco directorios y dentro de cada directorio crear 10 ficheros

- <https://www.jesusninoc.com/11/28/ejercicios-de-powershell-crear-cinco-directorios-y-dentro-de-cada-directorio-crear-10-ficheros/>

Crear una estructura de directorios

- <https://www.jesusninoc.com/11/28/ejercicios-de-powershell-crear-una-estructura-de-directorios/>

Crear una estructura de directorios (pero solo los números pares)

- <https://www.jesusninoc.com/11/28/ejercicios-de-powershell-crear-una-estructura-de-directorios-pero-solo-los-numeros-pares/>

Crear una estructura de directorios (pero solo los números impares)

- <https://www.jesusninoc.com/11/30/ejercicios-de-powershell-crear-una-estructura-de-directorios-pero-solo-los-numeros-impares/>

Crear directorios con los meses del año

- <https://www.jesusninoc.com/12/01/ejercicios-de-powershell-crear-directorios-con-los-meses-del-ano/>

Crear directorios con los años desde 2020 hasta 2030 (opción 1)

- <https://www.jesusninoc.com/12/02/ejercicios-de-powershell-crear-directorios-con-los-anos-desde-2020-hasta-2030-opcion-1/>

Crear directorios con los años desde 2020 hasta 2030 (opción 2)

- <https://www.jesusninoc.com/12/03/ejercicios-de-powershell-crear-directorios-con-los-anos-desde-2020-hasta-2030-opcion-2/>

Crear directorios con los años desde 2020 hasta 2030 (opción 3)

- <https://www.jesusninoc.com/12/04/ejercicios-de-powershell-crear-directorios-con-los-anos-desde-2020-hasta-2030-opcion-3/>
-

Más información

- <https://www.jesusninoc.com/07/04/4-gestion-del-sistema-de-archivos-en-powershell/>
-

Gestión de entrada y salida

El sistema operativo controla dispositivos de E/S, se encarga de capturar interrupciones, enviar y manejar datos memoria que recoge desde los dispositivos, etc.

Ejemplos

Ver los teléfonos móviles que se han conectado al sistema operativo

- <https://www.jesusninoc.com/2016/12/01/ver-los-telefonos-moviles-que-se-han-conectado-al-sistema-operativo/>

Conocer el estado del ratón

- <https://www.jesusninoc.com/2016/10/05/conocer-el-estado-del-raton/>

Activar un dispositivo Plug and Play con PowerShell

- <https://www.jesusninoc.com/2018/03/08/activar-un-dispositivo-plug-and-play-con-powershell/>

Desactivar un dispositivo Plug and Play con PowerShell

- <https://www.jesusninoc.com/2018/02/08/desactivar-un-dispositivo-plug-and-play-con-powershell/>

Más información

- https://www.jesusninoc.com/07/03/3-gestion-del-hardware-en-powershell/#Dispositivos_de_entrada_y_salida
-

Tipos de sistemas operativos

Una de las cosas que más ha evolucionado en el campo de la informática son los sistemas operativos, en este apartado vamos a clasificar los sistemas operativos atendiendo a varios criterios como son: el número de usuarios, el número de procesos, el número de sesiones, el número de procesadores y el sistema informático o el tipo de dispositivo utilizado.

El número de usuarios

Sistemas operativos monousuario

Un único usuario utiliza el sistema operativo al mismo tiempo. Algunos sistemas operativos conocidos de este tipo son Windows 7, Windows Vista, Windows XP, etc.

Sistemas operativos multiusuario

Varios usuarios acceden al sistema operativo y utilizan los recursos del ordenador simultáneamente, se puede realizar por

medio de terminales locales o remotas. El sistema operativo tiene que ser capaz de proteger el acceso entre ellos. Algunos sistemas operativos multiusuario son Windows Server, Linux, Unix, etc.

El número de procesos o tareas

Sistemas operativos monotarea o monoprogramación

Solamente se puede ejecutar un proceso a la vez. Un ejemplo de este tipo de sistema es DOS.

Sistemas operativos multitarea o multiprogramación

Varios procesos se ejecutan a la vez. La mayoría de los sistemas operativos actuales son de este tipo, algunos ejemplo son Unix, Linux, las versiones de Windows a partir de Windows NT como 2000 Profesional, XP, Vista, 7, etc.

El número de sesiones

Sistemas operativos monosesión

Sistemas operativos que son capaces de ejecutar una sola sesión de usuario. Los sistemas operativos de hace tiempo como Windows 98 o Windows 95.

Sistemas operativos multisesión

Sistemas operativos que son capaces de ejecutar varias sesiones de usuario a la vez. La funcionalidad de multisesión se puede instalar mediante aplicaciones, pero es el sistema operativo el que permite dicho funcionamiento. Los servidores de la familia Windows Server, la mayoría de las versiones de Linux.

El número de procesadores

Sistemas operativos monoprocesador

Solo hay un procesador en el ordenador. Los sistemas operativos que se tenían que ejecutar con un solo procesador eran los basados en DOS.

Sistemas operativos multiprocesador

Hay varios procesadores en el ordenador. Algunos ejemplos de sistemas operativos que admiten varios procesadores son los Windows a partir de NT, también la familia de Windows Server, Unix, Linux, etc.

El sistema informático o el tipo de dispositivo utilizado

Sistemas operativos para supercomputadoras y computadoras centrales (Mainframes)

Estos sistemas están orientados al procesamiento por lotes (procesamiento de trabajos sin que actúe un usuario), transacciones (muchas peticiones de poco tamaño) y tiempo compartido (varios usuarios ejecutando trabajos al mismo tiempo), la mayoría de los sistemas operativos para este tipo de sistemas están basados en Linux.

Sistemas operativos para servidores

Se encargan de dar servicios a través de la red, algunos sistemas operativos para servidores son Solaris, Linux, Windows Server, etc.

Sistemas operativos para ordenadores personales

Los sistemas operativos que se utilizan para este tipo de ordenadores tienen que dar buen soporte al usuario para realizar tareas básicas, hay multitud de sistemas operativos para ordenadores personales aunque los más conocidos son Linux, Windows, Mac OS, etc.

Sistemas operativos para dispositivos de bolsillo

Los sistemas operativos para este tipo de dispositivos cada vez son más sofisticados debido a que continuamente incorporan nuevas funcionalidades, algunos de los sistemas operativos más populares de estos dispositivos son Android, BlackBerry OS, iPhone OS, Palm WebOS y Windows Mobile.

Sistemas operativos para dispositivos integrados

También se conocen como sistemas embebidos o incrustados, estos dispositivos se encuentran integrados en televisiones, coches, móviles, etc. Para estos también hay sistemas operativos como por ejemplo VxWorks que se utiliza para controlar las centralitas electrónicas de los coches.

Sistemas operativos para tarjetas inteligentes

Este tipo de dispositivos permiten la ejecución de cierta lógica programada, algunas tarjetas sólo contienen una memoria no volátil en cambio otros tienen CPU y memoria volátil (RAM). Sirven para como medio de identificación, control de acceso, firma digital, sistema de pago, etc. Estos dispositivos también disponen de sistemas operativos, normalmente este sistema operativo se carga el proceso de fabricación y no puede ser modificado, aunque también existen excepciones con algunas tarjetas que permiten incorporar programas al sistema operativo una vez cargado.

Tipos de aplicaciones

Es la parte que no se puede tocar del ordenador (intangibile), el software es un elemento lógico y se define como un conjunto de órdenes e instrucciones que cuando se ejecutan sirven para realizar alguna tarea, los programas sin ejecutar son simples archivos en disco.

El software se puede dividir en tipos:

- Software de sistema. Conjunto de programas que administran los recursos del ordenador.
- Software de programación. Es el conjunto de herramientas que sirven para crear nuevos programas, éstos se crean utilizando un lenguaje de programación. Un lenguaje de programación es un conjunto de símbolos, reglas sintácticas y semánticas que indican cómo hacer un programa. Todas las órdenes o instrucciones que forman el programa se escriben en unos archivos llamados códigos fuente.
- Software de aplicación. Se refiere a los programas que permiten realizar tareas a los usuarios con conocimientos básicos de informática. Les permite sacar provecho al ordenador y así realizar tareas relacionadas con su actividad como, por ejemplo, escribir un documento, enviar un mail, etc.

Licencias y tipos de licencias

El licenciamiento se trata de un procedimiento (o un contrato mejor dicho) para conceder a una persona o entidad el derecho de usar un software con fines industriales, comerciales o personales, de acuerdo a las cláusulas que en ella aparecen.

Algunos tipos de licencias:

- Licencia de software de dominio público
- Licencia de software de semi libre
- Licencia de software libre
- Licencia de software de libre no protegido con copyleft
- Licencia de software de Copyleft
- Licencia de software de GPL
- Licencia de software de Debian
- Licencia de software de BSD
- Licencia de software de MPL y derivadas
- Licencia de software con copyleft

- Licencia de software de Freeware
- Licencia de software de Postcardware
- Licencia de software de Donationware
- Licencia de software de Shareware
- Licencia de software de Demo
- Licencia de software de Abandonware
- Licencia de software de X.org
- Licencia de software de fuentes abiertas / o código abierto
- Licencia de software de código abierto permisivas
- Licencia de software de código abierto robustas
- Licencias de software de código abierto robustas fuertes
- Licencias de software de código abierto robustas débiles
- Licencia de software de código cerrado
- Licencia de software de privativo
- Licencia de software de Comercial
- Licencia de software de OEM
- Licencia de software de Retail
- Licencia de software de volumen

Gestores de arranque

Un gestor de arranque (en inglés bootmanager) es un programa que permite elegir el siguiente código a ejecutar en el proceso de arranque, tradicionalmente a través de un menú.

Habitualmente el gestor de arranque forma parte del cargador de arranque como por ejemplo en GRUB, GRUB 2, LILO o SYSLINUX. Sin embargo, el cargador de arranque puede no tener gestor de arranque (ej. EFISTUB) y el gestor de arranque puede ser un programa independiente como por ejemplo rEFIt, rEFInd, fallback.efi o los gestores de arranque nativos UEFI.□

Más información

- https://www.jesusninoc.com/06/09/administracion-de-procesos-del-sistema-administracion-de-sistemas-operativos/#Secuencia_de_arranque_del_sistema_Demonios
-

Consideraciones previas a la instalación de sistemas operativos libres y propietarios

Antes de realizar una instalación hay que tener en cuenta algunos parámetros:

Lugar

- **Local:** La instalación se realiza en el mismo ordenador.
- **Red:** La instalación se realiza desde otro ordenador a través de la red.

Tipo

- **Instalación normal:** Durante el proceso de instalación se responde a una serie de preguntas relacionadas con la configuración del ordenador y los usuarios.
- **Actualización:** proceso mediante el cual pasamos de una versión de un sistema operativo a otra más actual, añadiendo características nuevas, manteniendo los datos y los programas que estaban instalados en el ordenador (puede ser que algunos programas no funcionen correctamente con el nuevo sistema operativo, entonces habrá que reinstalar). La mayoría de los sistemas

operativos permiten actualizar de una versión a otra superior de una forma rápida y sencilla. Antes de hacer una actualización hay que asegurarse que el ordenador cumple con los requisitos para soportar el nuevo sistema operativo.

- **Downgrade:** Consiste en instalar una versión anterior de un sistema operativo, esta operación se hace por ejemplo para que se puedan ejecutar determinadas aplicaciones que en versiones más actuales de un sistema operativo no funcionan correctamente.
- **Migración:** Proceso mediante el cual pasamos de un sistema operativo a otro (pueden ser sistemas operativo de la misma familia o no) sin conservar datos y programas que estaban instalados, la única forma de conservar los datos es haciendo una copia de seguridad de los datos en algún dispositivo de almacenamiento como un CD, DVD, dispositivo USB, tarjeta de memoria, etc.

Entorno

- **Virtual:** El sistema operativo se instala en un entorno virtual. La ventaja de virtualizar es que no necesita un gestor de arranque y se pueden utilizar dos sistemas operativos de forma simultánea sin necesidad de reiniciar el equipo.
- **No virtual:** El sistema operativo se instala en un entorno no virtual.

Atención

- **Atendida:** Es necesario estar delante del ordenador respondiendo a preguntas para configurar la instalación.
- **Desatendida:** No es necesario estar delante del ordenador respondiendo a preguntas, se utiliza un archivo de respuesta que contiene la información que necesita el

sistema operativo para instalarse correctamente. El fichero se puede pasar a través de un dispositivo USB.

Modo

- **Mediante ficheros:** De un modo u otro, el sistema operativo necesita ficheros para funcionar, cuando instalamos el sistema operativo lo que hacemos es copiar esos ficheros en el disco duro del ordenador.
- **Mediante imágenes:** Una imagen es un fichero o dispositivo de almacenamiento que contiene la estructura y toda la información de un disco duro, una partición, un dispositivo USB, un CD, etc. En el caso de una imagen para un disco duro se realiza creando una copia completa de sector por sector.

Forma

- **Mediante dispositivos:** USB, CD/DVD, disco duro, etc.
- **Carpetas compartidas:** Esta forma de instalación es típica de una red, un ordenador que ya tiene un sistema operativo previamente cargado, comparte un recurso en red.
- **PXE (Preboot eXecution Environment, Entorno de ejecución de prearranque):** Este entorno se basa en la comunicación de cliente-servidor, el ordenador que soporta PXE se comporta como un cliente que envía una petición al servidor solicitando la asignación de una IP, el servidor que tiene DHCP asigna una dirección IP al cliente, después el servidor envía un programa para arrancar la instalación utilizando el protocolo de transferencia TFTP (Trivial File Transfer Protocol, protocolo de transferencia de archivos trivial), el programa de arranque contiene la ubicación de los archivos necesarios para la instalación, entonces el

cliente comienza la descarga de esos archivos.

Instalación de sistemas operativos. Requisitos, versiones y licencias

Antes de instalar un sistema operativo hay que tener en cuenta una serie de pasos que se resumen en los siguientes verbos: planificar, preparar, ejecutar, configurar y documentar.

Planificación de la instalación

Planificar la instalación es algo fundamental y nos ayudará a realizar cualquier tipo de instalación.

Algunos puntos que tenemos que planificar son:

Compatibilidad del sistema operativo con el hardware del ordenador en donde se va a instalar

Ejemplo

- <https://www.jesusninoc.com/09/07/obtener-informacion-sobre-el-hardware-creando-un-objeto-con-todos-los-datos/>
-

La partición en donde se va a instalar el sistema operativo

Ejemplos

- <https://www.jesusninoc.com/06/10/sistema-operativo-insta>

[lado-sobre-una-particion/](#)

Ejercicios

- <https://www.jesusninoc.com/11/05/ejercicios-de-powershell-crear-varios-discos-virtuales-leyendo-desde-un-fichero-la-informacion-nombre-los-discos-hay-que-particionarlos-y-formatearlos-anadir-a-cada-disco-duro-informacion-sobre-l/>
-

Más información

- <https://www.jesusninoc.com/10/28/comandos-utiles-para-trabajar-con-discos-en-linux/>
 - <https://www.jesusninoc.com/07/04/4-gestion-del-sistema-de-archivos-en-powershell/#Discos>
-

Compatibilidad con las aplicaciones que se van a instalar y ejecutar

Licencias disponibles

Ejemplo

- <https://www.jesusninoc.com/12/02/obtener-serial-de-windows-con-powershell/>

Número de ordenadores a instalar

El sistema operativo previamente instalado en el equipo

Si tiene un sistema operativo instalado se puede realizar una actualización y si no tiene un sistema operativo instalado se realiza una **instalación limpia**.

Ejemplo

- <https://www.jesusninoc.com/06/10/sistema-operativo-instalado-sobre-una-particion/>
-

Preparar el ordenador para la instalación

Tener en cuenta desde dónde se carga el sistema, cuáles son los parámetros para realizar la instalación, en qué disco y partición se va a instalar el sistema, etc.

Ejecución de la instalación

Aunque el sistema de particionado y dar formato se puede realizar antes de ejecutar el programa instalador, la mayoría de los instaladores permiten realizar esas dos operaciones durante la instalación del sistema operativo.

Es muy importante leer bien las distintas pantallas que van apareciendo en el proceso de instalación.

Configuración posterior a la instalación

Lo primero que hay que configurar es el gestor de arranque dependiendo del sistema operativo que hayamos instalado habrá que configurar uno u otro.

Después hay que configurar los drivers, la red, las actualizaciones y el antivirus y por último personalizar el sistema operativo.

Ejemplo

- <https://www.jesusninoc.com/01/25/analizar-informacion-obtenida-sobre-las-actualizaciones-get-hotfix/>
-

Documentación de la instalación

Durante el proceso de instalación hay que documentar los pasos que se van realizando, señalando las incidencias que van surgiendo, documentar las cosas es una buena práctica y ayuda a otras personas y a nosotros a saber qué es lo que se ha hecho con un equipo o qué configuración se ha utilizado, también controlar los tiempos dedicados a cada paso y las incidencias en el caso de que aparezcan.

Ejemplos sobre instalaciones

Se trata de ejemplos posibles que habría que implementar físicamente.

Instalación del sistema operativo que se realiza al directivo de una empresa

Lugar	Tipo	Entorno	Atención	Modo	Forma
Local	Actualización	No virtual	Desatendida	Ficheros	USB

Instalación del sistema operativo que se realiza a los administradores de sistemas

Lugar	Tipo	Entorno	Atención	Modo	Forma
Red	Migración	Virtual	Atendida	Imágenes	Carpeta compartida

Instalación que se realiza al personal de recursos humanos

Lugar	Tipo	Entorno	Atención	Modo	Forma
Local	Normal	No virtual	Desatendida	Imágenes	PXE

Instalación/desinstalación de aplicaciones. Requisitos, versiones y licencias

Cuando se habla de software tenemos que tener en cuenta qué se puede hacer con el mismo:

- Instalar.
- Actualizar.
- Downgrade (devolver el software a una antigua versión).
- Desinstalar.

La instalación y desinstalación de programas es una tarea importante y se tiene que hacer con control, no hay que instalar un programa sin estar seguro de lo que hace, y siempre hay que conocer la fuente de donde proviene el programa.

Los sistemas operativos tienen instalado por defecto multitud de programas que nos permiten realizar gran cantidad de tareas como escribir documentos, escuchar música, navegar por Internet, etc. Sin embargo, esto no es suficiente para la

mayoría de los usuarios y es probable que necesiten instalarse nuevos programas como, por ejemplo, un navegador de Internet distinto al que tiene el sistema operativo por defecto, un conjunto de herramientas ofimáticas más completo, etc.

Aunque lo normal es instalar una sola vez un programa, a veces hay algún problema en el programa y es necesario reinstalar algún archivo o el programa entero.

Es importante tener el software correctamente instalado, identificado y controlar el acceso a los programas por parte de los usuarios.

Ejemplo

Leer de un fichero nombres de programas e instalarlos (ejercicio)

- <https://www.jesusninoc.com/03/11/ejercicios-de-powershell-leer-de-un-fichero-nombres-de-programas-e-instalarlos-2/>
-

Más información

- <https://www.jesusninoc.com/07/05/5-gestion-del-software-en-powershell/>
-

Actualización de sistemas

operativos y aplicaciones

Cada día aparecen nuevas vulnerabilidades en los sistemas operativos y en los programas, es importante mantener el sistema operativo actualizado con los últimos parches de seguridad.

Las actualizaciones sirven para evitar problemas o corregirlos, de esta forma el sistema operativo se mantiene seguro. Las actualizaciones son necesarias para los sistemas operativos.

Los sistemas de gestión de actualizaciones permiten que éstas se descarguen y se instalen con orden, de no ser así podrían aparecer problemas si el sistema operativo comenzara a descargar todas las actualizaciones a la vez, en algunos casos pueden llegar a saturar ciertos recursos como, por ejemplo, la conexión de red. Para resolver este problema algunos sistemas operativos tienen programas que descargan las actualizaciones y las envían a otros ordenadores a la red de una manera ordenada.

Gracias a los sistemas de actualizaciones, el tiempo que va desde que se registra un fallo hasta que se corrige es muy pequeño.

Ejercicios

Ejercicios sobre actualizaciones en PowerShell

- <https://www.jesusninoc.com/03/11/ejercicios-de-powershell-actualizaciones-en-powershell/>

Obtener información sobre las actualizaciones del sistema operativo

- <https://www.jesusninoc.com/03/11/ejercicios-de-powershell-obtener-informacion-sobre-las-actualizaciones/>
-

Ficheros de inicio de sistemas operativos

La secuencia de arranque de Windows utiliza el sistema de almacenamiento y configuración de arranque denominado BCD store (Boot Configuration Data). Este registro se encuentra situado en el directorio \boot\BCD de la partición donde se instala el sistema de forma oculta y en binario. Consiste en un repositorio de datos y parámetros necesarios para gestionar el arranque del sistema. El fichero Bootmgr.exe es el encargado del manejador y cargador de arranque (boot loader).

En la secuencia o proceso de arranque linux tras encender el ordenador, lo primero que hace la CPU es ejecutar el código máquina alojado en una dirección de memoria predefinida. Este código es el que representa a la BIOS que una vez que se haya cargado chequea los periféricos y localiza un dispositivo con el que arrancar el sistema, es decir, una de sus funciones es buscar un programa especial que ejecutar llamado gestor de arranque (GRUB). Esta técnica se llama bootstrapping, siendo así, un bootstrap o boot loader el programa que ejecuta a otro que no puede ejecutarse por sí mismo. De esta forma podemos decir que la BIOS es el bootstrap para gestor de arranque, que a su vez es el bootstrat del S.O a ejecutar.

Registro del sistema

El registro en Windows es una base de datos jerárquica central utilizada en Microsoft Windows, utilizada para almacenar información que es necesaria para configurar el sistema para

uno o más usuarios, aplicaciones y dispositivos de hardware.

El Registro contiene información a la que Windows hace referencia continuamente durante el funcionamiento, como los perfiles de cada usuario, las aplicaciones instaladas en la computadora y los tipos de documentos que cada uno puede crear, la configuración de la hoja de propiedades para las carpetas y los iconos de las aplicaciones, qué hardware existe en el sistema, y los puertos que se están utilizando.

Ejemplos

- <https://www.jesusninoc.com/04/13/ejecutar-solo-una-vez-un-programa-al-iniciar-la-sesion-de-un-usuario/>
-

Actualización y mantenimiento de controladores de dispositivos

Cada día aparecen nuevas vulnerabilidades en los sistemas operativos y en los programas, es importante mantener el sistema operativo actualizado con los últimos parches de seguridad.

Las actualizaciones sirven para evitar problemas o corregirlos, de esta forma el sistema operativo se mantiene seguro, las actualizaciones son importantes y necesarias para los sistemas operativos.

Los sistemas de gestión de actualizaciones permiten que las actualizaciones se descarguen y se instalen con orden, de no ser así puede suponer un problema si el sistema operativo comienza a descargar todas las actualizaciones a la vez y en

algunos casos pueden llegar a saturar ciertos recursos como por ejemplo la conexión de red, para resolver este problema algunos sistemas operativos tienen programas que descargan las actualizaciones y las envían a otros ordenadores a la red de una manera ordenada.