

Sistemas operativos

Autor: Katherine Roa Banquez



Sistemas operativos / Katherine Roa Banquez, / Bogotá D.C.,
Fundación Universitaria del Área Andina. 2017

978-958-5455-72-6

Catalogación en la fuente Fundación Universitaria del Área Andina (Bogotá).

© 2017. FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA
© 2017, PROGRAMA INGENIERIA DE SISTEMAS
© 2017, KATHERINE ROA BANQUEZ

Edición:

Fondo editorial Areandino
Fundación Universitaria del Área Andina
Calle 71 11-14, Bogotá D.C., Colombia
Tel.: (57-1) 7 42 19 64 ext. 1228
E-mail: publicaciones@areandina.edu.co
<http://www.areandina.edu.co>

Primera edición: noviembre de 2017

Corrección de estilo, diagramación y edición: Dirección Nacional de Operaciones virtuales
Diseño y compilación electrónica: Dirección Nacional de Investigación

Hecho en Colombia
Made in Colombia

Todos los derechos reservados. Queda prohibida la reproducción total o parcial de esta obra y su tratamiento o transmisión por cualquier medio o método sin autorización escrita de la Fundación Universitaria del Área Andina y sus autores.



Sistemas operativos

Autor: Katherine Roa Banquez





Índice

UNIDAD 1 Conceptos generales de los sistemas operativos

Introducción	7
Metodología	8
Desarrollo temático	9

UNIDAD 1 Fundamentos de los sistemas operativos

Introducción	16
Metodología	17
Desarrollo temático	18

UNIDAD 2 Gestión de procesos

Introducción	26
Metodología	27
Desarrollo temático	28

UNIDAD 2 Planificación de procesos

Introducción	37
Metodología	38
Desarrollo temático	39



Índice

UNIDAD 3 Gestión de memoria real

Introducción	49
Metodología	50
Desarrollo temático	51

UNIDAD 3 Gestión de memoria virtual

Introducción	61
Metodología	62
Desarrollo temático	63

UNIDAD 4 Gestión de ficheros

Introducción	73
Metodología	74
Desarrollo temático	75

UNIDAD 4 Gestión de Entrada/Salida y seguridad en sistemas operativos

Introducción	86
Metodología	87
Desarrollo temático	88

Bibliografía	95
--------------	----

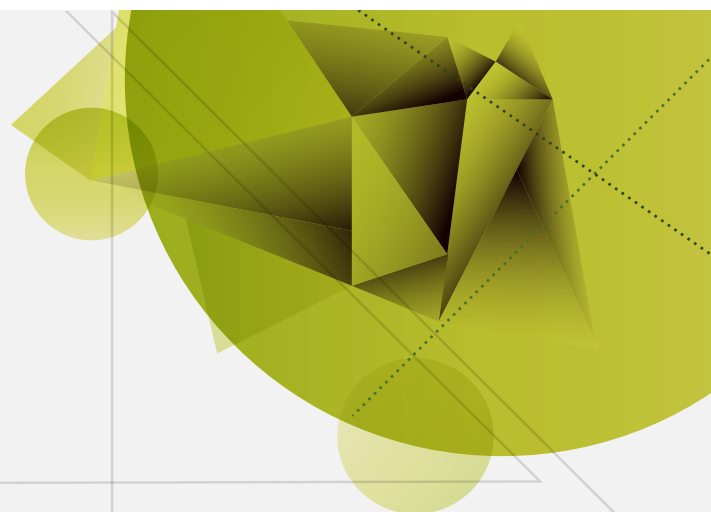


1

Unidad 1

Conceptos generales
de los sistemas
operativos

• • • •



Sistemas operativos

Autor: Katherine Roa

Introducción

El capítulo inicia con la definición de un Sistema Operativo (SO) abordado desde varios autores, se presenta un SO como un intermediario entre un sistema de cómputo y el usuario, cuya tarea es suministrar un buen servicio a estos y alcanzar un uso eficiente de este.

La función principal de un Sistema Operativo es admitir la ejecución de los programas del usuario para asegurar su conveniencia y el uso eficiente de los recursos. En esta guía describiremos las funciones realizadas por el Sistema Operativo para atender las necesidades de un usuario.

Continuaremos conociendo los objetivos y funciones de los sistemas operativos al igual que su evolución a lo largo del tiempo y finalizaremos con la estructura de un Sistema Operativo o mejor conocido como métodos de estructuración.

Para un mayor entendimiento de este capítulo, se recomienda al estudiante realizar una lectura de la guía y generar un reporte crítico del mismo a modo personal, con el fin de afianzar los conocimientos adquiridos en esta cartilla.

Se sugiere adicionalmente, desarrollar las diferentes lecturas y actividades planteadas en la semana, con el fin de comprender mejor la temática trabajada.

Conceptos generales de los sistemas operativos

Definición de los sistemas operativos

Si queremos conocer el significado de un Sistema Operativo (SO), tal vez, podamos encontrar diferentes respuestas, teniendo en cuenta a la persona a la que le preguntemos. En este sentido Alegre (2010) expone que:

- Para un joven universitario, un Sistema Operativo es el software que permite el acceso al conocimiento disponible en internet.
- Para un programador, un Sistema Operativo es el software que permite el uso de un sistema de cómputo para el desarrollo de programas.
- Para un técnico de planta informatizada, el Sistema Operativo es el componente de un sistema de cómputo que controla la planta.
- Para una persona del común (que usa un paquete de aplicaciones), un Sistema Operativo es simplemente el software que hace posible que use el paquete.

En nuestro caso nos vamos a enfocar en la definición dada por Candela, S., García, C. Quesada, A. Santana, F. Santos, J. (2007), "... es un programa que actúa como intermediario entre el usuario y el hardware de un

sistema de cómputo. El propósito de un Sistema Operativo es ofrecer un ambiente en el que el usuario pueda ejecutar programas de una forma cómoda y eficiente".

En consecuencia a lo anterior, el hardware de un computador no puede trabajar por sí solo, necesita de unos programas (software) que hacen que el sistema arranque y funcione, a este software que sirve de intermediario entre los usuarios y el hardware es lo que se conoce como Sistema Operativo.

Por lo tanto, un Sistema Operativo es quizás el componente más importante de una computadora, dado que cuando se carga el Sistema Operativo en la memoria y se ejecuta, este despierta el ordenador y hace que reconozca a la CPU, la memoria, las unidades de disco y cualquier dispositivo de entrada y salida, verificando su conectividad y eficiencia en sus procesos.

Un Sistema Operativo controla el uso de los recursos de un sistema de cómputo, tales como la CPU, memoria y los dispositivos de Entrada y Salida (E/S) a fin de satisfacer los requerimientos de los usuarios. Estos esperan conveniencia, calidad de servicio y seguridad al ejecutar sus programas, mientras que los administradores del sistema esperan un uso eficiente de los recursos de la computadora y un buen desempeño al ejecutar los programas de los usuarios.

La interacción del Sistema Operativo con el sistema de cómputo y los programas del usuario es un aspecto crucial de su operación. Las características de la arquitectura del sistema de cómputo se usan para realizar esta.

Un ambiente de cómputo se caracteriza por un sistema de cómputo, sus interfaces con otros sistemas, sus usuarios y la naturaleza de sus requisitos computacionales. Las metas de un Sistema Operativo se determinan por la noción de la utilización efectiva de un sistema de cómputo en el ambiente donde se usa, de modo que los sistemas operativos se clasifican sobre la base de los ambientes de cómputo donde se emplea.

Objetivos y funciones de los sistemas operativos

Se dice que un Sistema Operativo es manejado por eventos. Un evento es cualquier situación que requiere atención del Sistema Operativo. Por ejemplo, una petición del recurso por un programa del usuario o el fin de una operación de E/S. Cuando ocurre un evento, el control del CPU se pasa al Sistema Operativo. Este analiza el acontecimiento y realiza acciones apropiadas. Por ejemplo, cuando un programa pide recurso, el Sistema Operativo lo asigna si está disponible y cuando termina la operación E/S, informa al programa que solicitó la operación de E/S e inicia otra operación de E/S en el dispositivo, si alguno está pendiente. Después de atender el evento, el Sistema Operativo da las instrucciones a un programa de usuario para la ejecución en el CPU.

El funcionamiento manejado por eventos del Sistema Operativo mostrado en la Figura 1, es una visión lógica de su funcionamiento. En la visión física corresponde, el fin

de una operación de E/S o una solicitud del recurso pedido por un programa causa una interrupción en el sistema de cómputo. El CPU se diseña para reconocer una interrupción y desviarse a sí mismo a la ejecución de una rutina de interrupción del proceso, que activa una rutina apropiada de manejo de los eventos.



Figura 1. Un Sistema Operativo en su ambiente
Fuente: Propia.

Un Sistema Operativo no solo debe asegurar el uso eficiente de un sistema de cómputo, sino que también debe proveer la conveniencia del usuario. La meta crucial de un Sistema Operativo es proveer la combinación que mejor se ajuste al ambiente de uso eficiente y de conveniencia del usuario, a lo cual se le llama utilización efectiva.

El uso eficiente del sistema es la mayor preocupación en un ambiente de computación que se centra en el sistema, tal como el procesamiento de datos no interactivo. Esto se obtiene a través del uso de buenas políticas de asignación de recursos.

La eficiencia de uso tiene dos aspectos, un Sistema Operativo consume algunos recursos de un sistema de cómputo durante su propia operación; por ejemplo, ocupa memoria y usa CPU. Este consumo de recursos

constituye una sobrecarga que reduce los recursos disponibles a los programas de los usuarios.

El otro aspecto del uso eficiente tiene que ver con el empleo de recursos que realizan los programas del usuario. Puede resultar una eficiencia baja debido a dos causas; si un Sistema Operativo le asigna exceso de recursos a los programas o si es incapaz de asignar recursos libres a los programas que los necesite. Lo primero conduce a un derroche de recursos, y el otro produce entorpecido de los recursos y afecta los progresos de los programas.

Encontramos un gran número de sistemas operativos en uso porque cada uno de ellos provee un gusto diferente de utilización efectiva; por un lado, podemos encontrar sistemas operativos que proveen un servicio rápido requerido por las aplicaciones de comandos y de control, por otro lado, tenemos sistemas operativos que hacen un uso

eficiente de los recursos de la computadora para proveer computación de bajo costo.

Un Sistema Operativo implementa requisitos computacionales de sus usuarios con la ayuda de recursos del sistema de cómputo. Sus funciones cruciales son:

Programa: iniciación y terminación de los programas. Proporcionar métodos beneficiosos de modo que varios programas puedan trabajar con el objetivo común.

Recursos: asegurar la disponibilidad de los recursos del sistema y asignarlos a los programas.

Planificación: decidir cuándo y durante cuánto tiempo dedicará la CPU a un programa.

Protección: proteger los datos y los programas contra la interferencia de otros usuarios y de sus programas.

Evolución histórica



Imagen 1. Evolución de los sistemas operativos

Fuente: <https://sistemasoperativosuvn.files.wordpress.com/2013/02/linea-del-tiempo.jpg>

En la imagen 1. Se observa un resumen de la evolución de los sistemas operativos en cada una de las décadas, iniciando desde la década de los 40 con la creación de la primera generación de computadoras en el año 1940, estas computadoras funcionaban con válvulas o tubos al vacío, se accedía directamente a la consola por medio de una serie de micros interruptores que permitían introducir el programa en la memoria del computador; estas eran utilizadas en el ámbito científico y militar.

En el año 1951, las computadoras de esta generación, usaban transistores para procesar información, estas computadoras ya trabajaban con lenguajes de programación “de alto-nivel”; uno de los primeros lenguajes fueron COBOL y FORTRAN. Los sistemas operativos fueron bastante simples, con conceptos tales como el monitor residente, el proceso por lotes y el almacenamiento temporal.

En 1964, con esta generación se inició el uso de los circuitos integrados, este tipo de computadoras trabajan con terminales remotas, las cuales permiten acceder a la computadora central para realizar operaciones, también podían ser conectadas a Internet. Los sistemas operativos trabajaban técnicas de multiprogramación, multiprocesador, tiempo compartido, entre otros. Los lenguajes utilizados fueron el FORTRAN, ALGOL y COBOL.

En la década de los 70, en el año 1970, se da inicio UNIX, y con el aparecen los sistemas operativos multiusuario - multitarea, Multiplexed information and computing service), el cual implementó un único nivel de almacenamiento para el acceso a los datos.

En la década de los 80, en el año 1981, nace el Sistema Operativo MS-DOS (Micro Soft Disk Operating System), creado por Microsoft para IBM PC. Desde esta fecha iniciaron con las diferentes generaciones, desde la versión PC DOS 1.0.

En el año 1984, nace el Sistema Operativo Mac OS cuya característica principal era una GUI (Graphic User Interface), multitareas y mouse; a finales de la década, para el año 1987 surge el Sistema Operativo OS/2 de IBM que intentó reemplazar a DOS como Sistema Operativo de las computadoras personales, era un SO multitarea, reconocía múltiples aplicaciones ejecutándose a la vez, sin embargo solo podía mostrar una aplicación a la vez en la pantalla.

En el año 1990, nacen dos grandes Sistemas operativos, SunOS y BeOS, el primero, fue la versión del SO derivado de Unix y BSD desarrollado por Sun Microsystems para sus estaciones de trabajo y servidores, y el segundo, fue desarrollado por Be Incorporated, orientado principalmente a proveer alto rendimiento en aplicaciones multimedia.

En 1992, sale al mercado el Sistema Operativo Solaris de tipo Unix desarrollado por Sun Microsystems y actualmente por Oracle Corporation, en esta misma década en el año 1993 nace Windows NT, el cual pertenece a la familia de sistemas operativos producidos por Microsoft, de aquí en adelante lanzan las diferentes versiones de Windows: Windows 98 (1998), Windows 10 (Beta), Windows Server (2000), Windows XP (2001), Windows Vista (2007), Windows 7 (2009) y Windows 8.x (2012).

En el año 2001, nace el Sistema Operativo MAC OS X, el cual está basado en el entorno operativo Unix, este SO es desarrollado, comercializado y vendido por Apple Inc.

Durante los años 2005 – 2010, emerge OpenSolaris como Sistema Operativo libre, a partir de la versión privativa de Solaris de Sun Microsystems, a partir de esta generación han nacido otras versiones como IllumOS, OpenIndiana.

Métodos de estructuración

A continuación se presenta cinco diseños de métodos de estructuración:

- Sistemas monolíticos
- Sistemas en capas
- Máquinas virtuales
- Microkernels

Sistemas monolíticos

Es una de las estructuras más utilizadas, teniendo en cuenta que no requiere de una

estructura, el Sistema Operativo funciona como una recopilación de procedimientos, donde cada una tiene una interfaz definida desde el punto de vista de parámetros y resultados, los cuales pueden hacer uso de otros procedimientos.

Otra característica fundamental del sistema monolítico, es que trabaja una bajo una estructura básica, para explicar este proceso, nos enfocaremos en lo expuesto por el autor Tanenbaum (2003).

1. Un programa principal que invoca el procedimiento de servicio solicitado.
2. Un conjunto de procedimientos de servicio que ejecutan las llamadas al sistema.
3. Un conjunto de procedimientos utilitarios que apoyan a los procedimientos de servicio.

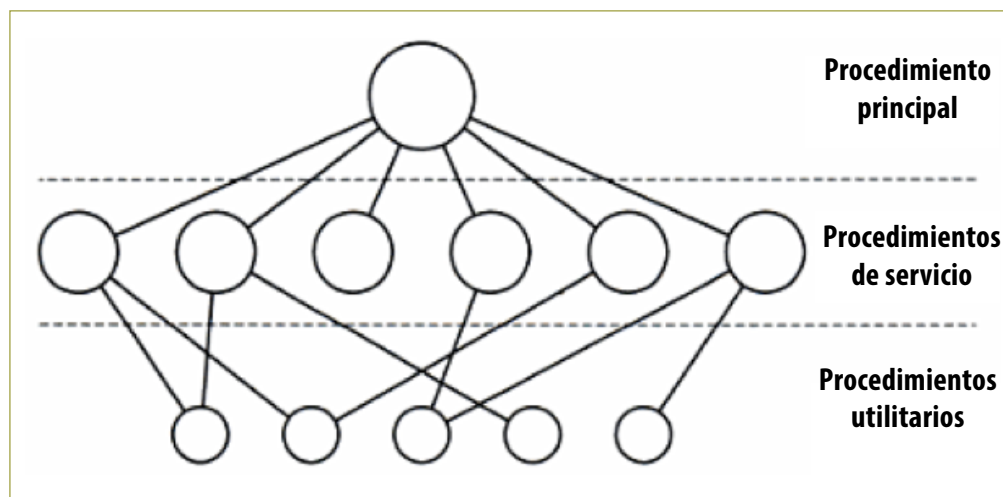


Imagen 2. Modelo de estructuración para un sistema monolítico
Fuente: Tanenbaum (2003).

Sistemas en capas

Este método consiste en estructurar el Sistema Operativo de forma jerárquica entre capas, cada una fijada por la que está abajo.

Como ejemplo de este método, vamos a referenciar el expuesto por el autor Tanenbaum (2003), el cual hace referencia al THE (Techische Hogescool Eindhoven), creado por E. W. Dijkstra y sus estudiantes en el año 1968 en los países bajos. Este sistema trabaja por lotes y cuenta con seis capas:

Capa 0: asignación del procesador y multiprogramación: es el encargado de asignar el procesador, conmutado entre procesos al presentarse interrupciones o expirar temporizadores. Esta capa es la que hace posible la multiprogramación de la CPU.

Capa 1: administración de memoria y tambor: es la encargada de repartir espacios a los procesos en la memoria principal y el tambor.

Capa 2: comunicador operador-proceso: es la encargada de la comunicación entre cada proceso y la consola del operador.

Capa 3: administración de entrada/salida: se encarga de administrar los dispositivos de entrada y salida.

Capa 4: programas de usuario: es la encargada de administrar los procesos, memoria, dispositivos E/S que se ejecutaban desde los programas.

Capa 5: el operador: es el proceso operador de THE.

Máquinas virtuales

Este tipo de método se basa en tiempo compartido que proporciona multiprogramación y una máquina extendida como una interfaz más cómoda. El monitor de la máquina virtual quien es el corazón del sistema, se ejecuta en el hardware y realiza la multiprogramación proporcionando no solamente una máquina virtual sino varias, las cuales son copias exactas del hardware de las máquinas reales.

Microkernel

Este método consiste según Tanenbaum (2003) "...en asignar recursos a las máquinas virtuales y luego examinar cualquier intento de usarlos para garantizar que ninguna máquina use los recursos de otra. Cada máquina virtual en nivel de usuario puede ejecutar su propio Sistema Operativo".