

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

ARAGÓN

“ADMINISTRACIÓN BÁSICA DEL SISTEMA OPERATIVO SOLARIS 2.7”

T E S I S

Que para obtener el Título de:

no (*acompañado de un Disco Compacto*)

INGENIERO EN COMPUTACIÓN

Presenta:

ALEJANDRO JIMÉNEZ VARGAS

Asesor: Ing. Juárez Ortega Israel

México D., F. Octubre de 2001



TESIS CON FALLA DE ORIGEN



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A MIS PADRES

Gracias Padre por ser para mí el más grande ejemplo a seguir,
por toda la paciencia que tuviste y tienes conmigo,
por todos aquellos consejos y motivación que siempre he recibido de ti,
por todo el apoyo que me diste y que todavía me brindas;
y sobre todo

Gracias por ser el mejor de los Padres ya que si no fuera así no sería lo que ahora soy.
Gracias Papá.

Gracias Madre por todos tus cuidados y tiempo dedicado durante tantos años,
por todos los consejos que siempre me diste,
por esos regaños que me diste y que solo ahora comprendo,
por darme siempre lo mejor de ti;
y sobre todo

Gracias por ser la mejor de las Madres ya que si no fuera así no sería lo que ahora soy.
Gracias Mamá.

A MIS HERMANOS

Gracias Hermanos por ser un gran ejemplo a seguir para mí,
por resolver mis dudas cuando se me presentaban,
por ese amor que sabemos existe en nosotros aunque nos es difícil expresar,
por esa paciencia que siempre han tenido conmigo,
por el apoyo que me dieron y continúan dándome;
y sobre todo

Gracias por ser los mejores Hermanos que pude haber tenido
Gracias Hermanos.

A ADRIANA Y FERNANDO

Gracias por todo el apoyo que me han dado,
por cooperar en la realización de este trabajo, sin ustedes hubiera sido más difícil,
por ser para mí una motivación que me hace intentar superarme día a día,
por estar junto a mí en estos días tan difíciles;
y sobre todo

Gracias Adriana por ser una compañera que siempre me ha dado lo mejor de sí y
Gracias Fernando por ser un hijo maravilloso
Gracias.

A LA MAESTRA CUCA

Gracias Maestra por haberme dedicado parte de su tiempo,
por haberme enseñado las primeras letras y números,
por su atención tan especial y afectiva que siempre tuvo conmigo,
por esos regaños y jalones de cabellos que me dio, de algo me sirvieron;
y sobre todo

Gracias por haber hecho los cimientos de mi educación escolar

Es muy difícil para mí poder expresar todo el amor, y agradecimiento que siento por ustedes en unas cuantas líneas, sin su constante apoyo, paciencia y amor no habría podido alcanzar esta meta o tal vez me hubiera sido extremadamente difícil alcanzarla. Sé que este es solo el primer paso de muchos que hay que dar y que el camino siguiente también es muy difícil, sólo espero seguir contando siempre con ustedes.

Alejandro

ÍNDICE

ÍNDICE

INTRODUCCIÓN.....	I
-------------------	---

I Conceptualización de sistema operativo.

1.1 Definición de sistema Operativo.	1
- Objetivos de los sistemas operativos.	2
- Funciones de los sistemas operativos.	2
1.2 Evolución de los sistemas operativos.	3
- Pasos de trabajo.	3
- Controladores de dispositivos.	4
- Proceso por lotes.	5
- Secuencia automática de trabajos.	5
- Estados de ejecución.	6
- Interrupciones.	7
- Spooling.	7
- Multiprogramación.	8
- Tiempo compartido.	9
1.3 Estructura de los sistemas operativos.	9
- Sistema Monolítico.	10
- Sistemas en Estratos o Jerárquico.	11
- Máquinas Virtuales.	12
- Servidores.	12
- El modelo cliente servidor para estaciones de trabajo en red.	13
- Clientes en red.	14
Resúmen del Capítulo I.	15

II Sistema Operativo UNIX.

2.1 Introducción del sistema operativo UNIX.	17
2.2 Características del sistema operativo UNIX.	17
- Código fuente abierto.	17
- Herramientas y utilidades cooperativas.	17
- Capacidades multiusuario y multitarea.	18
- Entorno excelente para redes.	18
- Portabilidad.	18
- Aplicaciones.	18
- Utilidades.	19
2.3 Nacimiento del sistema operativo UNIX.	19
2.4 Factores que influyeron en el desarrollo de UNIX.	21
- El sistema XENIX.	22

TESIS CON
FALLA DE ORIGEN

2.5 UNIX Sistema V Versión 4.	23
- Estándares.	23
- La definición de la interfaz del Sistema V.	23
- POXIS.	24
- X/OPEN.	24
2.6 Características de la Versión 4.	25
- El conjunto unificado de órdenes.	25
- Shells.	25
- El árbol de directorios.	25
- Entrada/Salida y acceso al sistema.	26
- Procesamiento en tiempo real.	26
- Administración del sistema.	26
- Conexión en red.	27
- Interfaces de usuario.	27
2.7 Variantes del Sistema Operativo UNIX.	28
- UNIXWARE.	28
- UNIX SCO.	28
- IRIX.	29
- HP-UX.	29
- AIX.	29
- LINUX.	29
- A/UX.	29
2.8 Estructura del sistema operativo UNIX.	30
- Organización de la información.	30
- Concepto de archivo.	31
- Concepto de directorio.	31
- El sistema de archivos.	32
- El Shell.	34
- El núcleo.	35
2.9 Sistema operativo Solaris.	35
2.10 Interfaz de usuario.	36
2.11 Novedades de la versión de Solaris 2.7.	37
2.12 Características del nuevo sistema operativo Solaris 2.7.	38
- Sistema operativo de 64 bits.	38
- Navegador de Web.	38
- Administración de red y del sistema.	38
- Rendimiento de red.	39
- Seguridad de red.	39
- Instalación.	40
- Documentación.	40
- Soporte de idioma.	40
- Estándares.	41
- Gráficos/Imágenes.	41
- Escritorio.	41
- Impresión.	42
- Entorno de desarrollo de software.	42
Resumen de capítulo II.	43

III Administración del Sistema Operativo Solaris 2.7.

3.1 Responsabilidades y labores del administrador.	46
3.2 Instalación.	48
3.3 Instalación del hardware.	48
3.4 Conectar los componentes del sistema.	49
3.5 Conexión física.	49
3.6 Open Boot PROM.	51
- Elementos básicos.	51
- Contenido del Boot PROM.	51
- Comandos básicos del Boot PROM.	52
- El comando banner.	53
- El comando boot.	53
- El comando help.	54
- Cambio del dispositivo de inicialización por default.	54
- El bus SCSI.	55
- Tarjeta de direcciones en el ambiente Sun.	56
3.7 Instalación del sistema operativo.	57
- Grupo de software.	58
- Configuración del clusters.	59
- Requerimientos de hardware.	60
- Información para la identificación del sistema.	60
- Preparando la instalación.	61
3.8 Administración.	82
3.9 Encendido del equipo.	83
3.10 Pasos del Boot PROM.	83
- El archivo /etc/inittab.	85
- Arranque y desconexión.	87
3.11 El Comando init.	87
3.12 Apagado.	88
- El comando shutdown.	88
3.13 Archivos de administración.	89
- El sistema de archivos raíz.	89
- El sistema de archivos /home.	90
- El sistema de archivos /usr.	90
- El sistema de archivos /var.	90
3.14 Instalación de software.	91
- El comando pkginfo.	91
- El comando pkgrm.	92
- El comando pkgadd.	93
- El comando pkgchk.	94
- El archivo /var/sadm/install/contens.	94
- Los paquetes spooling.	95
3.15 Instalación de parches.	95
3.16 Procesos del sistema.	98
- El comando ps.	98
- Opciones del comando ps.	99
- El comando kill.	100

- Procesos padre e hijos.	101
- Demonios.	101
- Procesos zombies.	101
- El comando at.	102
- El comando crontab.	103
- El comando pgrep y pkill.	104
- El comando nohup.	105
3.17 Adición de usuarios (useradd).	105
- Opciones del comando useradd.	106
- Adición de un grupo (groupadd).	107
- Variables de shell.	107
- El comando cp y mv.	109
- El editor vi.	111
3.18 Seguridad.	114
- Archivos de contraseña.	114
- El archivo /etc/passwd.	114
- El archivo /etc/shadow.	115
- El archivo /etc/group.	117
- El archivo /var/adm/sulog.	118
- El archivo /etc/default/passwd.	118
- El archivo /etc/default/login.	119
- El archivo /etc/default/su.	120
- El archivo /etc/hosts.equiv y el archivo .rhosts.	120
- Permisos de archivo.	121
- El comando chmod.	122
- El comando umask.	123
- El bit sticky.	124
- Niveles de seguridad de sistema operativo.	125
- Protección mínima.	125
- Protección de seguridad discrecional.	125
- Protección de acceso controlado.	125
- Protección de seguridad etiquetada.	125
- Protección estructurada.	126
- Dominios de seguridad.	126
- Diseño verificado.	126
3.19 Adición de una impresora.	126
- Comandos básicos de lp.	131
- El comando lp.	132
- El comando lpstat.	133
- El comando cancel.	134
3.20 File Systems.	134
- Raw y partición de bloque.	135
- La etiqueta del disco (VTOC).	135
- El Bootblock.	135
- El Superblock.	135
- Respaldo del Superblock.	136
- Grupo de cilindros.	136
- El comando fsck.	137
- Montado de los file systems.	142

- El archivo vfstab.	144
3.21 Administración de discos	146
- Nombres de dispositivos.	147
- El comando dmesg.	149
- El comando format.	149
- Reconfiguración de dispositivos.	150
- Particiones de disco y el comando format.	151
- El archivo format.dat.	159
3.22 Respaldos.	160
- El comando ufsdump.	162
- El comando ufsrestore.	165
- El comando tar.	169
- El comando cpio.	170
- El comando mt.	171
- Pasos para adicionar un disco al sistema.	171
3.23 Algunos comandos de administración avanzada.	172
- El comando iostat.	172
- El comando vmstat.	174
- El comando swap.	175
- El comando netstat.	176
- El comando snoop.	177
- El comando mpstat.	178
- El comando sar.	178
Resúmen del capítulo III.	181

IV Introducción a NetBackup 3.2.

Introducción.	185
4.1 Dispositivos de almacenamiento.	185
- Discos flexibles.	185
- Discos duros.	186
- Cintas magnéticas.	186
- CD-ROM.	187
- DVD.	188
- Cartuchos de cinta DLT.	189
- Guía general de manejo de los cartuchos DLT.	189
- Procedimiento de inspección visual del mecanismo.	190
- Inspección visual de un cartucho de cinta DLT.	190
4.2 Copias de seguridad.	191
- ¿Por qué hacer respaldos?	192
4.3 Sun StorEdge Enterprise NetBackup.	193
- Características de NetBackup.	194
- NetBackup servidor y clientes.	194
- Sun StorEdge Enterprise NetBackup: Manejador centralizado.	194

4.4 Introducción a la librería Sun Storeage L11000.	195
- Características de la librería.	195
- Componentes de la librería.	196
4.5 Ambiente.	197
- Características del servidor Maestro (mxscbcon01).	199
- Características del servidor esclavo (mxscbcal01).	199
- Características del servidor esclavo (mxscbbd01).	199
- Características de la librería L11000.	200
- ¿Cómo iniciar NetBackup 3.2?	200
- Política de respaldo.	209
- Asignación lógica con la física.	212
Resumen de Capítulo IV.	216

Conclusiones.

Bibliografía.

INTRODUCCIÓN

INTRODUCCIÓN

El principal objetivo de este trabajo es dar a conocer los aspectos fundamentales para la administración básica del sistema Operativo Solaris 2.7, mediante un texto práctico que pudiera ser leído por estudiantes y egresados de la carrera de Ingeniería en computación (ICO) como un manual de este tema para fortalecer sus conocimientos sobre UNIX Solaris y la administración básica en general.

Son diversas las razones que tuve para escoger y realizar esta investigación, en primer lugar, el darme cuenta de lo importante que es tener conocimientos sobre estos temas y la ausencia de materias o materiales que permitan al estudiante o recién egresado de ICO incorporarse a trabajos afines a él, o bien le resulte más difícil conocer este sistema y adentrarse en el mercado laboral.

Mi experiencia al egresar de la carrera fue descubrir la necesidad de contar con conocimientos sobre los diferentes sistemas operativos y equipos informáticos en el mercado, pues de ello depende una adecuada colocación laboral para implementar los conocimientos adquiridos en la escuela.

Así, al tener contacto con equipos Sun y conocer sus ventajas y características, decidí realizar un trabajo de tesis que abordara este tema y pudiera servir de documentación a alumnos y egresados de ICO, de modo que contaran con mayor conocimiento y posibilidades de desarrollo en su vida profesional.

Sin embargo, al comenzar a buscar información para documentar este trabajo, me di cuenta de la dificultad de conseguir información en español, clara y utilizable sobre el sistema operativo Solaris, lo que refuerza mi idea de la necesidad de un manual introductorio sobre este sistema para quienes, como yo, se interesen en conocerlo y aplicarlo en su ejercicio laboral.

Otra de las razones para elegir este tema, fue darme cuenta de que el sistema operativo Solaris es muy utilizado en el mercado laboral, sobre todo en las grandes empresas mexicanas, como PEMEX, Telmex, CFE, SHCP, ISSSTE, BIMBO, etcétera, y son cada vez más el número de empresas (pequeñas y medianas) que implementan, en menor escala, este sistema.

Por ello, es importante que el alumno de ICO cuente por lo menos con los conocimientos de administración básica sobre Solaris, pues esto le ampliará su abanico de posibilidades laborales, así como permitirle el conocimiento de tecnología de punta y tener nociones aplicables a otros equipos y sistemas.

Por todo lo anterior, decidí realizar una investigación sobre este sistema, presentando una visión general sobre su administración, pues sería imposible realizar un solo manual sobre toda la información relacionada con él. De manera que este trabajo es solamente una introducción a la administración del sistema operativo Solaris 2.7, basándome en mi experiencia laboral como administrador Solaris.

Aunque los temas que presento son abordados desde una perspectiva general, mi interés es exponer conocimientos básicos y prácticos para que el recién egresado de ICO empiece a conocer y manejar la administración de este sistema y pueda después ir adquiriendo otros conocimientos con la práctica en el manejo del mismo y la documentación que vaya obteniendo en su desarrollo profesional.

Los temas que abordo en este trabajo son una visión general de las tareas más frecuentes que debe realizar un administrador del sistema operativo Solaris, pues cada uno de ellos es tan basto que podría ser el tema particular de una tesis completa. Por esto, el contenido de mi investigación sólo se refiere a las tareas habituales que un administrador a nivel básico debe realizar.

Este trabajo es una propuesta de manual teórico-práctico del sistema Solaris 2.7 enfocado a los estudiantes avanzados o recién egresado de ICO, que tienen conocimientos de este sistema operativo o de UNIX, por lo que un alumno que no ha tenido relación con él podría carecer de cierta información para comprender este material.

El primer capítulo, *Conceptualización del sistema operativo*, hace una descripción general de la evolución de los sistemas operativos, definiéndolos y especificando sus objetivos, funciones, desarrollo, estructura y características, presentando las nociones elementales para comprender el modo y funcionamiento de un sistema operativo en general.

El capítulo II *-Sistema Operativo UNIX-* se aboca a estudiar el sistema operativo UNIX en particular, haciendo una breve introducción a este sistema, presentando sus características, evolución, aspectos que influyeron en su desarrollo, variantes y estructura, además muestra el sistema operativo Solaris y su versión 2.7, con sus características y ventajas sobre otros sistemas operativos.

El capítulo III *-Administración del Sistema Operativo Solaris 2.7-* es el centro de esta investigación, en él se define desde qué es la administración del sistema, hasta las responsabilidades y labores de un administrador de Solaris, así como algunos de los comandos básicos que éste debe conocer para garantizar un buen funcionamiento del sistema.

El último capítulo *-Introducción a NetBackup-* es un tema anexo para describir una de las principales labores del administrador de Solaris: los dispositivos de almacenamiento y las medidas de respaldo de la información, además presenta un ejemplo real describiendo el funcionamiento de NetBackup.

Estos temas están desarrollados para que los posibles lectores conozcan las principales tareas del administrador del sistema operativo Solaris, de modo que este trabajo podría emplearse como un manual de introducción para la administración de Solaris o ser un documento de consulta para las personas interesadas en alguno de los temas mencionados.

CAPÍTULO I

TESIS CON
FALLA DE ORIGEN

I. CONCEPTUALIZACIÓN DE SISTEMA OPERATIVO

1.1 Definición de sistema operativo

“El sistema operativo es un conjunto de programas del sistema que controla los recursos del ordenador y sirve de base para la escritura y ejecución de los programas de aplicación”¹.

Podemos decir que el sistema operativo es un conjunto de programas que actúa de intermediario entre el hardware de una computadora y los usuarios de la misma, permitiendo un uso eficaz y fácil del sistema.

Cuando un usuario necesita utilizar la computadora, requiere una serie de recursos que han de ser gestionados adecuadamente para obtener el máximo rendimiento; el sistema operativo actúa entonces como manejador de los recursos disponibles planificando, mediante una política adecuada, su utilización más eficaz. (Fig. 1.1)

Las computadoras modernas constan de grandes memorias, procesadores complejos y una inmensa red de dispositivos periféricos que van desde una simple pantalla a impresoras de rayos láser o conexiones de red de diversas unidades centrales; todo ello requiere de una perfecta armonía entre los procesos que utilizan el sistema. Los sistemas operativos son los encargados de armonizar estos procesos, repartiendo los recursos disponibles entre los usuarios.

Así, el sistema operativo ha de controlar los dispositivos de entrada y salida, supervisando a su vez la ejecución de los programas de usuario.

Aparece pues una doble función del sistema operativo; por una parte, como gestor de los recursos de la computadora y por otra, como controlador de los programas y de su movimiento entre la unidad central y los procesos. Pero un sistema operativo se justifica más por lo que hace que por lo que es. Los sistemas operativos nacen como elemento necesario para hacer viable la utilización de recursos físicos necesarios para resolver ciertos programas de aplicación.

En resumen, se puede definir el sistema operativo como el conjunto de programas de control y programas de proceso que, relacionados entre sí, dan coherencia a las acciones que lleva a cabo una computadora optimizando el uso de la misma mediante un aprovechamiento eficaz de todos los recursos y una adecuada comunicación entre usuario y computadora.

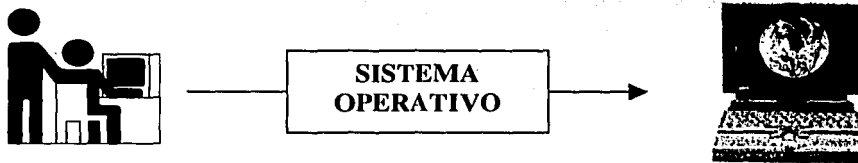


Fig. 1.1 Relación máquina usuario.

¹ Juanes Baza, David. *Sistemas operativos* p. 15

Objetivos de los sistemas operativos

Un sistema operativo es un programa que controla la ejecución de programas de aplicación y actúa como una interfaz entre el usuario y el hardware de una computadora. Puede considerarse que un sistema operativo tiene varios objetivos principales.

- Optimizar el rendimiento y el uso del sistema.
- Administrar los recursos del hardware y hacer más accesible a los usuarios lo relacionado a sus programas.
- Mayor productividad del usuario.
- Adaptabilidad de los programas a los cambios de los componentes del sistema.
- Posibilidad de extensión de las capacidades funcionales del sistema operativo de manera que satisfagan circunstancias cambiantes.
- Tiempos de respuesta mejorados que satisfagan más fácilmente los requerimientos de tiempo real.
- Aumentar la eficiencia y con ello disminuir el costo del uso del sistema.
- Mejorar el uso del hardware al optimizar la corriente del trabajo y realizar decisiones sobre el manejo de los recursos del sistema.

Funciones de los sistemas operativos

Un sistema operativo se encarga de crear el entorno donde se van a ejecutar los programas; por ello sus funciones están encaminadas a que dicha ejecución sea lo más rápida y fiable posible.

El sistema operativo actúa como un verdadero administrador de recursos y, como tal, debe realizar las funciones de asignar el recurso, controlar quién lo posee en cada instante y por último, recuperarlo.

Las funciones que ha de realizar un sistema operativo son:

- Aceptar los procesos y mantenerlos hasta su conclusión.
- Asignar y desasignar recursos.
- Interpretar los lenguajes de control.
- Contabilizar el uso de recursos.
- Detectar y corregir errores.
- Controlar las operaciones de E/S.
- Proteger los datos y los programas.
- Gestionar las interrupciones.
- Facilitar la interacción con el usuario.

1.2 Evolución de los sistemas operativos

Las primeras computadoras estaban compuestas exclusivamente por el sistema físico que se reducía a una computadora de grandes dimensiones a la que había que dotar de todas las instrucciones para que ejecutase un trabajo.

No existía un sistema operativo propiamente dicho. Los usuarios ponían su nombre y tiempo que necesitaban utilizar la máquina en una lista de espera. Durante el tiempo de utilización, cada usuario disponía de todos los recursos y empleaba la ejecución paso a paso para poner a punto sus programas. Toda la memoria era utilizada por el usuario (Fig. 1.2).

Este modo de operación llamado de “puerta abierta” era poco económico pues la inactividad del CPU, esperando decisiones del programador, podía alcanzar hasta el 90 % del tiempo. Además si se agotaba el tiempo asignado, el programador debía abandonar la unidad sin concluir el trabajo y, por el contrario, si terminaba antes de lo previsto, la computadora quedaba inactiva hasta que llegara el siguiente usuario.

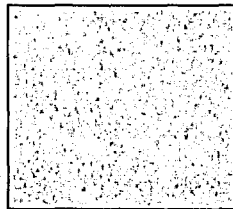


Fig. 1.2 Memoria limpia.

Pasos de trabajo

Para realizar un programa o un conjunto de programas, el usuario tenía que escribir el programa en el papel y después perforar las tarjetas que contenían dicho programa, posteriormente, llevar las tarjetas a la sala de la computadora para introducir en el sistema el contenido de las mismas.

Cuando la máquina terminaba con el programa había que tomar de la impresora la salida del mismo para comprobar, y corregir en su caso, los datos de la salida.

Sí además eran necesarios programas auxiliares como el compilador, estos deberían tomarse de la biblioteca de programas y leerse. Así el programador debía seguir las etapas o pasos de trabajo siguientes para ejecutar un programa.

- ◆ Escribir el programa en lenguaje de alto nivel.
- ◆ Perforar las correspondientes tarjetas e introducirlas en el dispositivo lector de las mismas.
- ◆ Montar el carrete de cinta donde se encontraba el traductor de lenguaje de alto nivel al lenguaje ensamblador.
- ◆ Ordenar la lectura y compilación del programa.

- ◆ Montar el carrete con el programa ensamblador que traduce de lenguaje ensamblador a lenguaje máquina.

Toda esta serie de pasos consumía una gran cantidad de tiempo al que se le denominaba “tiempo de preparación” (Fig. 1.3).

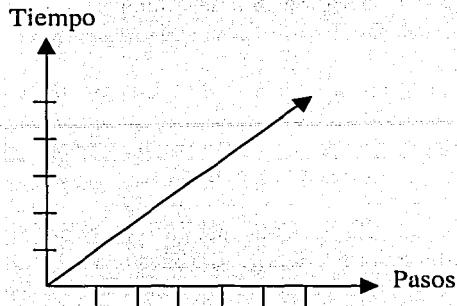


Fig. 1.3 Entre más pasos mayor el tiempo.

Controladores de dispositivos

Con el paso del tiempo, aparecieron dispositivos de E/S mucho más perfeccionados, como impresoras, lectoras de tarjetas, cintas magnéticas, etc.

Para manejar con aprovechamiento dichos dispositivos, el programador tuvo que añadir a sus trabajos los programas necesarios para controlar cada uno de los dispositivos que iba a utilizar.

Se adoptó el criterio de grabar en la memoria los programas específicos encargados de manejar cada dispositivo y tenerlos siempre a disposición de todos los usuarios.

A cada uno de estos programas se le llamó controlador de dispositivo (device driver). El esquema de la memoria era como muestra la figura 1.4.

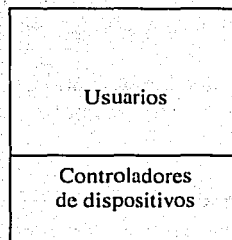


Fig. 1.4 Aspecto de la Memoria con los primeros controladores de dispositivos.

Proceso por lotes

“Para reducir el tiempo de preparación se recurrió a la agrupación de los trabajos que poseían características similares o que necesitaban utilizar los mismos recursos lógicos. Surge así el concepto de ‘trabajo por lote’ o ‘procesos BATCH”.

“El operador reunía las tarjetas correspondientes a trabajos similares, transfiriendo luego la información de dichas tarjetas a una cinta magnética”.

“La unidad de cinta que leía la cinta magnética estaba conectada a la máquina que recibía la información correspondiente”.

“Una vez que era procesado el lote, se transmitían los resultados a otra cinta que se instalaba en un carrete que mediante una unidad especializada transmitía los datos sobre una impresora”².

Pero aún con este sistema, se demoraban mucho las ejecuciones de los programas, ya que el operador debía cambiar los carretes, ordenar las ejecuciones, recoger los resultados, etc.

Secuencia automática de trabajos

Se confeccionaron programas que permitiesen la secuencia automática de trabajos. Estos programas se incorporaron a la memoria de la computadora y tomaron la mayoría de las decisiones que antes estaban encargadas a los operadores.

Pero todavía era necesario indicar a la computadora donde comenzaba cada trabajo. Para ello, se colocaron tarjetas con características especiales que marcaban el principio y el fin de cada programa; la computadora sólo tenía que leer e interpretar dichas tarjetas.

A estas tarjetas se las llamó tarjetas de control y al programa que las interpretaba se le denominó “lenguaje de control de trabajos”.

Las órdenes del lenguaje de control de trabajos incluyen tanto marcas de principio y final de trabajo, como órdenes para cargar y ejecutar los programas, necesidades de memoria y tiempos de ejecución entre otros conceptos.

Para que la secuencia de trabajo fuese totalmente automática había que incluir programas que actuasen en caso de una interrupción del proceso, bien por un error o por terminación del mismo. En la memoria deberían estar los siguientes programas: controladores de dispositivos, programas de gestión de errores y el programa intérprete del lenguaje de control.

² ídem. p. 22-23

La memoria tenía el aspecto mostrado en la figura 1.5

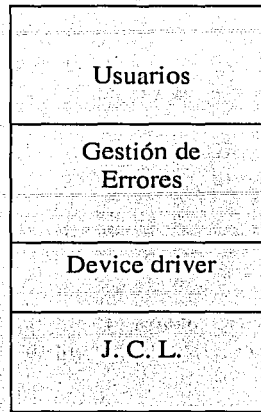


Fig. 1.5. Aspecto de la memoria con controladores de dispositivos, programas de gestión de errores y el programa intérprete del lenguaje de control.

Estados de ejecución

El conjunto de programas del que se ha hablado antes constituye por sí mismo un sistema operativo básico, pero aún se requieren unas funciones que lo perfeccionen.

Primeramente es necesario que los programas de usuario no puedan manipular libremente los dispositivos de E/S pues ello podría producir un colapso o bloqueo de los mismos. Además los usuarios han de tener prohibida la alteración del propio sistema operativo.

”Para conseguir estos dos objetivos, existe un bit llamado de control, mediante el cual se establecen dos modos o estados de ejecución, el estado monitor y el estado usuario.

Para protección del propio sistema operativo se comparan todas las direcciones de memoria a las que pretende acceder el usuario con las direcciones que ocupa el sistema operativo y si alguna coincide, se rechaza la petición del usuario, preservando así el acceso y posible alteración de los programas del sistema”³.

Cuando los programas de usuario necesitan utilizar dispositivos de E/S deben pedirselo al monitor mediante instrucciones denominadas llamadas al sistema. Una vez que concluye el servicio de E/S requerido, el control del sistema retorna al procesador.

³ Ídem. p.24

De todo lo anterior se deduce que mientras se está atendiendo una E/S el procesador permanece inactivo; por ello, es conveniente que las operaciones de entrada y salida se realicen lo más rápidamente posible.

Como los dispositivos de E/S son mucho más lentos que el procesador, se requiere un mecanismo para evitar las largas esperas de la unidad central.

Este mecanismo consiste en almacenar temporalmente la información que envía la unidad central y posteriormente dársela poco a poco a los dispositivos periféricos. Así la unidad central se ve pronto libre para realizar otros trabajos.

Este mecanismo se conoce como memoria tampón o buffer y el sistema se conoce con el nombre de entradas-salidas bufferadas.

Interrupciones

Con el sistema de buffer se consigue un importante ahorro de tiempo; sin embargo, para que dicho sistema sea eficaz, es necesario conocer lo antes posible cuando un dispositivo de E/S ha terminado una operación y puede comenzar con la siguiente.

Este problema se resuelve con el servicio de interrupciones. Tan pronto como un dispositivo de E/S termina una operación, interrumpe el CPU y esta cede el control a un programa residente en memoria con el nombre de servicio de interrupción que se encarga de iniciar la siguiente operación de E/S.

Spooling

Cuando se utilizaban tarjetas perforadas y cintas magnéticas, el principal problema consistía en que la lectora de tarjetas no podía escribir en un extremo de la cinta mientras el CPU leía del otro. Así que se grababan las cintas desde las tarjetas y después de ejecutar los programas en el CPU y de grabarlos en cinta, se pasaban los resultados de la cinta a una impresora. Se utilizaban las cintas y las impresoras de forma independiente al CPU es decir, de forma off-line.

Los discos eliminan estas dificultades, ya que se puede pasar moviendo la cabeza lectora de una parte del disco donde se han grabado tarjetas a otra donde el CPU puede leer otras tarjetas. Este sistema de operación simultánea se conoce con el nombre de spooling o spool (simultaneous peripheral operation on-line) y, esencialmente lo que hace es utilizar el disco como si se tratase de un gran buffer. Nada impide a esta forma de ejecución procesar un trabajo mientras se está imprimiendo otro.

Con el sistema buffering la E/S de un trabajo era simultánea a su propia ejecución. Con el spooling se puede simultanear entradas y salidas de un proceso con la ejecución de otro.

El spooling proporciona beneficios a las prestaciones de la computadora, pues permite trabajar al CPU y a los dispositivos periféricos con un rendimiento mucho mayor.

Al ser posible elegir los trabajos entre los residentes en un disco, el CPU puede planificar la forma y el orden de ejecutar dichos trabajos con vistas a obtener un mayor rendimiento.

Multiprogramación

La posibilidad del CPU de poder planificar los trabajos que están en el disco, proporciona otro tipo de ventajas adicionales.

“Cuando un trabajo en ejecución ha de detenerse para efectuar una E/S o por cualquier suceso, el CPU ya no permanece inactiva sino que toma otro trabajo y lo ejecutará, y si este también se detiene tomará otro o volverá al primero si es requerida para ello.

“Los métodos anteriores ejecutaban sólo un trabajo cada vez, lo que se conoce como monoprogamación . Ahora, varios procesos están siendo ejecutados simultáneamente y el proceso se conoce como multiprogramación”⁴.

La multiprogramación exige sistemas operativos mucho más avanzados ya que al existir varios programas en memoria se requiere una planificación del CPU para que elija el programa adecuado en cada momento así como planificación de los dispositivos que van a atender a los diferentes trabajos. (Fig. 1.6).

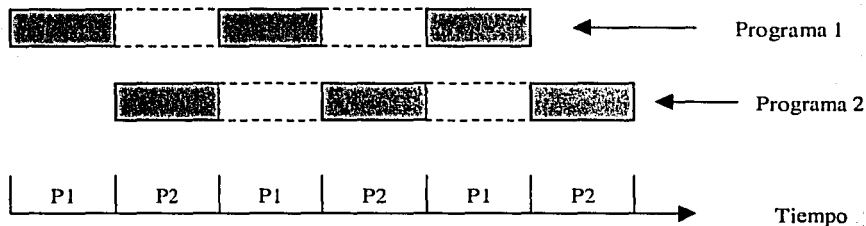


Fig. 1.6 Ejecución Multiprogramada.

⁴ Ídem. p.26

Tiempo compartido

La idea de trabajos por lotes no está en contradicción con la multiprogramación pero con la aparición de los discos se hizo posible el acceso inmediato a todos los trabajos. La aparición de nuevas aplicaciones proporcionaba a los sistemas la interacción entre trabajo y usuario lo cual estaba en contradicción con los trabajos en lote en los que el trabajo se preparaba, se dejaba en la máquina y posteriormente se recogían los resultados.

Para eliminar esta contradicción aparecieron los sistemas de tiempo compartido. Estos sistemas resuelven los problemas de modificaciones de programas a medida que se van ejecutando, y la obtención de resultados inmediatos, permiten dar respuestas rápidas.

Los sistemas de tiempo compartido consiguen un funcionamiento interactivo a bajo costo de inactividad del CPU.

Un sistema operativo de tiempo compartido permite a varios usuarios compartir simultáneamente el uso de la computadora mediante la planificación del CPU y la multiprogramación.

Aunque la idea de tiempo compartido data de 1960, el costo tecnológico que requiere hizo que dicho sistema no se difundiera hasta los años setentas.

1.3 Estructura de los sistemas operativos

“Un sistema operativo gestiona las operaciones de una computadora o de un sistema de computadoras y proporciona un número muy grande de servicios a los usuarios. Uno de los problemas con que se enfrentan los diseñadores de S. O, es el manejo de funciones tan complejas y a la vez tan fiables, y fáciles de manejar por los usuarios”⁵.

Cuantas más características se han añadido a un sistema operativo y a medida que el hardware básico se ha vuelto más complejo y versátil, han aumentado el tamaño y complejidad de los sistemas operativos.

El tamaño de un sistema operativo, caracterizado por complejo y la dificultad de la tarea que dirige, han conducido a tres problemas infortunados pero bastante comunes. Primero, de manera crónica, los sistemas operativos tardan en entregarse. Esto favorece los nuevos sistemas operativos y las mejoras de sistemas viejos. Segundo, los sistemas tienen defectos latentes que se muestran en el campo y deben repararse y volverse a trabajar. Y, por último, con frecuencia el rendimiento no es lo que se esperaba.

⁵ Ídem. p.30

Para manejar la complejidad de los sistemas operativos y para enfrentar estos problemas, todo el tiempo se ha prestado atención a la estructura del software del sistema operativo. El software debe ser modular. Esto ayuda a organizar el proceso de desarrollo del software y limita la tarea de diagnosticar y corregir errores. Los módulos deben tener interfaces bien definidas entre si y las interfaces deben ser tan simples como sea posible. De nuevo, esto facilita la tarea de programación. También hace más fácil la tarea de la evolución del sistema. Con interfaces mínimas limpias entre módulos, un módulo puede cambiarse, con impacto menor, a otro(s).

A lo largo de la historia se han seguido diversos procedimientos para la implementación de los sistemas operativos. Se describirán los más relevantes.

Sistema Monolítico

Esta es la organización más común, es un método que no exhibe estructura. El sistema operativo se escribe como un conjunto de procedimientos, cada uno de los cuales puede llamar a cualquiera de los otros siempre que lo necesite. Cuando se emplea esta técnica, cada procedimiento del sistema tiene una interfaz bien definida en términos de parámetros y resultados, y cada una tiene la libertad de llamar a cualquier otra, si ésta ofrece algún cálculo útil que la primera necesite.

Para construir el programa objeto real del sistema operativo cuando se usa este método, se compilan todos los procedimientos individuales o archivos que contengan los procedimientos y después se combinan todos en un solo archivo objeto con el enlazador. Casi no existe ocultamiento de información pues todo procedimiento es visible para todos. Aun en sistemas monolíticos es posible tener cuando menos una pequeña estructura mediante una instrucción que permite tener la máquina funcionando en dos modos: el modo usuario y el modo supervisor o modo Kernel (núcleo).

En el modo usuario no se pueden manejar todas la instrucciones del S. O. Lo que permite una protección del propio sistema. Cuando es necesario realizar alguna función que puede afectar el sistema, hay que hacer una llamada para que la computadora actúe en modo supervisor en el cual el sistema operativo controla la máquina. Una vez realizada la instrucción correspondiente, se devuelve el control al usuario para que continúe la ejecución del programa.

Una estructura para estos sistemas consiste en tener un programa central que llame al procedimiento del servicio solicitado, un conjunto de procedimientos para llamar al sistema y el conjunto de procedimientos de uso general para los diversos servicios. (Fig. 1.7)

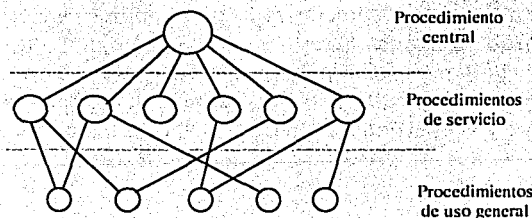


Figura 1.7 Estructura de un sistema monolítico.

Sistema en Estratos o Jerárquico

“Consiste en organizar el sistema operativo como una serie de estratos cada uno de los cuales interactúa con su entorno a través de interfaces bien definidas”.

“Cada problema complejo se divide en módulos simples, cada uno de los cuales realiza una función única. Los módulos son pequeños y fáciles de comprender, diseñar y cambiar en caso necesario”.

“En cada nivel de la jerarquía se ignoran los detalles de actuación de los niveles inferiores, produciendo un principio de ocultamiento de información que favorece la compartición multiusuario con independencia de los datos de cada uno de los usuarios del sistema”.

“El primer sistema construido en forma jerárquica fue el THE (Technische Hogeschool Eindhoven) construido por E. W. Dijkstra”⁶. Dicho sistema tenía seis niveles”.

En general, los niveles de todos los sistemas jerárquicos suelen tener la misma estructura:

El primer nivel suele corresponder al núcleo que contiene un planificador para seleccionar el trabajo que se ejecutará a continuación y gestiona las interrupciones.

El segundo, tercero y cuarto niveles se encargan de E/S, gestión de memoria y sistema de almacenamiento externo.

Los siguientes niveles constituyen el interfaz entre el sistema operativo y los usuarios. (Fig. 1.8)

La principal dificultad de este enfoque es la propia definición de los niveles ya que cada uno de ellos sólo puede utilizar los inferiores.

5	Operador de THE
4	Programas del usuario
3	Administración de entrada/salida
2	Comunicación entre el operador y el proceso
1	Administración de la memoria y el tambor
0	Distribución del procesador y multiprogramación

Fig. 1.8 Estructura del sistema operativo THE.

⁶ Tanenbaum, Andrew. *Sistemas Operativos Diseño e Implementación* p.40

TESIS CON
FALLA DE ORIGEN

Máquinas Virtuales

“Una máquina virtual consiste en un mecanismo que crea el sistema operativo de forma que una sola máquina real aparenta funcionar como varias máquinas independientes cada una de las cuales es utilizada por uno o varios usuarios.”⁷ (Fig. 1.9)

Para ello se comparten los recursos de la computadora de forma que cada usuario crea estar utilizando su propio procesador o su propia memoria.

El centro del sistema se conoce como monitor de la máquina virtual y se implementa en el hardware realizando multiprogramación con la creación de un estrato superior formado por varias máquinas virtuales que son copia exacta de la máquina real. Así cada máquina virtual puede disponer de su propio S. O. Y estos pueden ser diferentes para cada usuario.

Al hacer una separación completa de las funciones de multiprogramación y ofrecer una máquina extendida, cada una de las piezas puede ser mucho más simple y flexible.

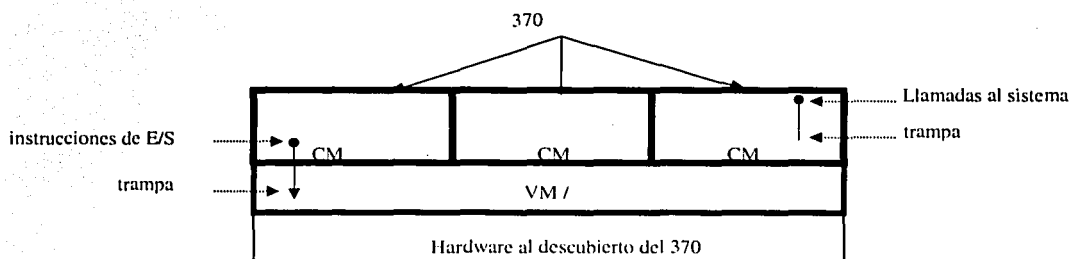


Fig. 1.9 Estructura de una Máquina Virtual (VM).

Servidores

“Una tendencia de los sistemas operativos modernos consiste en desplazar código a estratos superiores, pero en mayor proporción, y eliminar lo más que sea posible del sistema operativo, dejando un **kernel** mínimo. El método general consiste en implementar la mayoría de las funciones del sistema operativo en procesos del usuario. Para solicitar un servicio, como la lectura de un bloque de un archivo, un proceso del usuario envía la solicitud a un **proceso servidor** que realiza el trabajo y le devuelve una respuesta”⁸.

⁷ Juanes. Op. Cit. p. 32

⁸ Tanenbaum Op. Cit. p. 43

En este modelo todo lo que el kernel hace es manejar la comunicación entre clientes y servidores. Al dividir el sistema operativo en partes, cada una de las cuales se encarga del manejo de una faceta del sistema, como el servicio del archivo, el servicio del proceso, servicio de la terminal o el servicio de la memoria, cada parte se vuelve pequeña y difícil de manejar. Además, como todos los servidores corren como procesos en modo de usuario, y no en modo de kernel, no tienen acceso directo al hardware. Como consecuencia, si se activa un error oculto en el servidor, el servicio al archivo puede fracasar, pero generalmente no hará fallar a toda la máquina.

Otra ventaja del modelo del servidor del cliente es su adaptabilidad para utilizarse en sistemas distribuidos. Si un cliente se comunica con un servidor mediante la emisión de mensajes, el cliente no necesita saber si el mensaje se maneja en forma local o en su máquina, o bien si se envió a través de una red a un servidor en una máquina remota. Hasta donde concierne al cliente, sucede lo mismo en ambos casos: se envió una solicitud y se devolvió una contestación.

Para solicitar un servicio el proceso del usuario envía una solicitud a un proceso llamado servidor que realiza el trabajo y da la respuesta al usuario. El núcleo sirve entonces de enlace entre los usuarios y los servidores (Fig. 1.10).

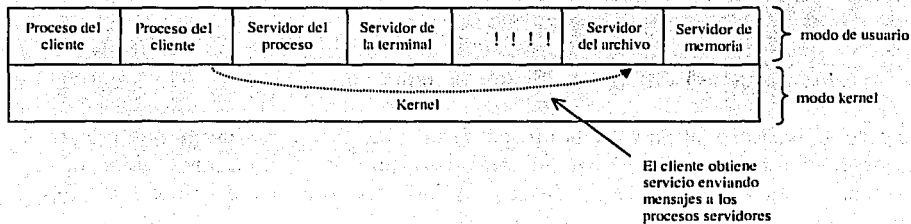


Fig. 1.10 Modelo del servidor del cliente.

El modelo cliente servidor para estaciones de trabajo en red

Un servidor es un host que provee servicios a otras máquinas en la red. Hay servidores de aplicación, de arranque, de instalación, de base de datos, de correo, de directorios compartidos, de licencia, de impresión, de servicio de nombres. Del término servidor se infiere que el sistema ofrece file systems y servicios a otros hosts en la red referidos como clientes.

Los clientes pueden ser también host que tienen un file system completo y no necesitan un servidor para operar. Un buen ejemplo de este tipo de cliente es un host que hace uso de una aplicación o servidor de impresión.

Algunos tipos de servidores son:

Servidores de aplicación: Proveen acceso de red a aplicaciones tales como FrameMaker, Photoshop, Lotus.

Servidores de arranque: Responden con parámetros de arranque a las diskless autoclient, JavaStation, e instalaciones de clientes de arranque desde la red.

Servidores de instalación: Posibilitan el acceso a Network File System (NFS) a la versión de software para instalación de clientes.

Servidores de Base de Datos: Ofrecen acceso a base de datos a través de aplicaciones tales como Oracle, Sybase, e Informix.

Servidores de correo: proveen transferencia Inter-networking de e-mail desde la red local.

Servidores de directorio Home: Exportan directorios home para el acceso de usuarios.

Servidores de licencia: Corren un demonio de licencia que administra el uso de licencias en un sistema para cada aplicación específica.

Servidores de impresión: Ofrecen servicios de impresión a sistemas en red. Las impresoras pueden ser ubicadas localmente o proveer servicios de spool a una impresora que tenga una dirección IP.

Cientes en red

La siguiente es una lista de los tipos de clientes en red.

Cliente diskless: no tiene almacenamiento en disco y es funcional únicamente a través del uso de memoria local e interfaz en red. Carga su configuración a la hora del arranque y usa memoria local para almacenar aplicaciones activas. Si la memoria es mínima, el cliente diskless accederá un archivo swap desde su servidor a través de la red.

Autocliente: Es similar al cliente diskless, tiene almacenamiento en disco local el cual se usa para el área de swap y para el cache usa su file system root. Puede acceder este file system root usando el cache si el servidor no esta disponible.

Javastation: Fue diseñado para "administración cero". Toda la administración incluyendo arranque, configuración y almacenamiento de datos es manipulado por servidores de red locales.

Standalone: No depende de un servidor para su funcionamiento. Aunque puede ocurrir que necesite acceder a un servidor con el propósito de correr una aplicación.

Existen numerosos sistemas operativos cada uno con características y beneficios propios.

RESUMEN CAPÍTULO I

En este capítulo se definió el Sistema Operativo como un conjunto de programas que planifica los recursos disponibles de la computadora, controlando y armonizando los dispositivos de entrada y salida (E/S), así como un controlador de los programas disponibles y el movimiento entre la unidad central y los procesos.

Los objetivos del Sistema operativo son: optimizar el uso y rendimiento del sistema, administrar los recursos de hardware y hacer más accesible a los usuarios lo concerniente a programas, permitiéndoles mayor productividad, adaptabilidad de los programas a los posibles cambios del sistema, mejorar los tiempos de respuesta, reducir el costo de uso del sistema y tomar decisiones sobre el manejo de los recursos del sistema en tiempo real.

El principal objetivo del sistema operativo es administrar adecuadamente los recursos, pues es el encargado de crear un entorno donde se ejecuten rápida y confiablemente los programas de la computadora. Sus funciones básicas son: asignar, controlar y recuperar tales recursos para facilitar la interacción entre el usuario y la máquina.

Existe gran variedad de sistemas operativos, con ventajas y características particulares, los cuales han evolucionado a lo largo de la historia moderna incorporándose a nuestra realidad. La evolución del sistema operativo ha estado encaminada a optimizar el uso de recursos, simplificar y reducir el tiempo de trabajo de los usuarios de la máquina, hasta llegar a la interacción inmediata que existe en nuestros días entre los individuos y las computadoras.

Al describir la evolución de la cibernética y el perfeccionamiento de los dispositivos de E/S, se recapitulan los convenientes e inconvenientes de tales avances, y como, para aprovecharlos al máximo, se hizo necesario que el programador añadiera nuevos programas de control para cada uno de ellos (controlador de dispositivos). Surge así la idea de grabar en la memoria de la computadora los programas correspondientes al manejo de tales dispositivos, de modo que siempre se encontraran a disposición del usuario.

Con el fin de priorizar la reducción del tiempo de preparación, surge la agrupación de trabajo con características afines o que empleen los mismos recursos lógicos, el llamado "trabajo por lotes" o "procesos BATCH". Después se desarrollaron programas de secuencia automática que, al incorporarse a la memoria de la máquina, se apropiaron de gran parte de las decisiones que antes le correspondían al operador. La memoria de la computadora debía contener tres programas básicos: de controladores de dispositivos, de gestión de errores y de intérprete de lenguaje de control.

Posteriormente, a fin de proteger al sistema operativo de posibles alteraciones de los programas del sistema y el bloqueo de dispositivos de E/S, se introdujo un bit de control que impedía la manipulación de dispositivos de E/S y la alteración del sistema operativo, estableciendo dos estados de ejecución: de monitor y de usuario. El control del sistema obedece a instrucciones concretas del usuario conocidas como llamadas al sistema.

Con la aparición de los discos se eliminaron muchas de las dificultades del sistema de tarjetas perforadas, pues permite una operación simultánea de lectura y grabación de CPU a disco, lo que se conoce como spooling y que permite la ejecución simultánea de instrucciones de procesos de E/S.

Por otro lado, está la posibilidad de que el CPU planifique los trabajos del disco con la multiprogramación, que permite que sean ejecutados simultáneamente varios procesos y que el CPU ya no esté inactivo, aprovechando así el tiempo y los recursos disponibles.

El surgimiento de nuevas aplicaciones permitió la interacción entre trabajo y usuario; aunque también implicó ciertas contradicciones entre estas innovaciones y las primeras aplicaciones, para disolverlas se crearon los tiempos compartidos, que resuelven los problemas de modificaciones de programas conforme son ejecutados, obteniendo resultados inmediatos.

Gracias a todas las innovaciones que surgieron en la evolución de los sistemas operativos, ha sido posible conseguir los avances tecnológicos con que cuentan las computadoras modernas. En el tema 1.3 se vió la estructura y algunas características de los actuales sistemas operativos que, al gestionar las operaciones de una computadora y proporcionar un gran número de servicios a los usuarios, implican una contradicción para los operadores: manejar funciones complejas, fiables y fáciles para los usuarios al mismo tiempo.

Esto, que conlleva múltiples ventajas, presenta también dificultades, las más comunes para los usuarios son: retardo en la entrega de los sistemas operativos, defectos en éstos y demora en su reparación, así como un rendimiento deficiente. Estos problemas han precisado la atención a la estructura de software en buscando sus soluciones; así, ha surgido el desarrollo de un software modular que optimiza los procesos, pues cada módulo cuenta con interfaces simples, definidas y limpias que facilitan la programación.

Existen diferentes procedimientos para implementar el sistema operativo. El sistema monolítico es el método de organización más común, pues no exhibe estructura, y contiene un conjunto de procedimientos con una interfaz con parámetros y resultados definidos. El sistema en estratos o jerárquico es otro de estos procedimientos, el cual se organiza según ciertos estratos que interactúan con el entorno mediante interfaces específicas.

Una característica de los sistemas operativos es la capacidad de crear máquinas virtuales, es decir, con la máquina real se puede aparentar tener otras independientes, para lograrlo, se utiliza la multiprogramación y se comparten los recursos de la máquina de modo que supone que cada usuario utiliza un procesador, sistema operativo y memoria individuales, aunque cada máquina virtual sea una copia idéntica de la real.

Otra característica son los procesos servidores, que son procesos de división del sistema operativo encargados de manejar aspectos determinados del sistema en procesos de modo de usuario; estos servidores tienen una estructura pequeña y de difícil manejo y no tienen acceso directo al hardware; se encargan de recibir las solicitudes de servicio de los usuarios y enviarles las respuestas a cada una de éstas.

Se puede definir un servidor como un proveedor de servicios a otras computadoras en red, que ofrece al cliente file systems y servicios a otros hosts en la red; puede ser de aplicación, de arranque, de instalación, de base de datos, de correo, de directorios compartidos, de licencia, de impresión, etcétera. Mientras que los clientes en red pueden ser de los siguientes tipos: diskless, autocliente, javastation, standalone, entre otros.

Las nociones presentadas en este capítulo son las elementales para entender a grandes rasgos el modo y funcionamiento del sistema operativo y adentrarse en un sistema operativo en particular.

CAPÍTULO II

II. SISTEMA OPERATIVO UNIX.

2.1 Introducción del sistema operativo UNIX

Durante los últimos veinte años el sistema operativo UNIX se ha convertido en un sistema operativo potente, flexible y versátil. Sirve como sistema operativo para todo tipo de computadoras, incluyendo las computadoras personales monousuario y las estaciones de trabajo de ingeniería, y microcomputadoras multiusuario, minicomputadoras, mainframes y supercomputadoras. El número de computadoras que funciona con este sistema operativo ha crecido de forma exponencial, con 10 millones aproximadamente de computadoras bajo UNIX y más de 50 millones de personas utilizando estos sistemas. El éxito del sistema UNIX se debe a muchos factores, entre los que se incluyen su portabilidad a un gran abanico de máquinas, su adaptabilidad y simplicidad, el amplio rango de tareas que puede ejecutar, su naturaleza multiusuario y multitarea y su adecuación a redes, que ha ido creciendo en importancia conforme ha prosperando Internet, por ello se espera que continúe con su rápido crecimiento.

2.2 Características del sistema operativo UNIX

Código fuente abierto

El código fuente del sistema UNIX, y no el código ejecutable, ha estado disponible a usuarios y programadores. A causa de esto, mucha gente ha sido capaz de adaptar el sistema de formas diferentes. Este carácter abierto ha conducido a la introducción de un amplio rango de características nuevas y de versiones personalizadas que se ajustan a necesidades específicas. A las personas que han desarrollado esta adaptación del sistema les ha resultado fácil, pues el código correspondiente es sencillo, modular y compacto. Esto ha fomentado la evolución del sistema, haciendo posible la fusión de las capacidades desarrolladas por diferentes variantes del sistema UNIX, necesarias para soportar los entornos de computación actuales en un sistema operativo único.

Herramientas y utilidades cooperativas

El sistema UNIX proporciona a los usuarios diferentes herramientas y utilidades que se pueden utilizar para realizar una gran variedad de trabajos. Algunas de ellas, son órdenes simples que se pueden utilizar para llevar a cabo tareas específicas; otras son pequeños lenguajes de programación que se pueden utilizar para construir guiones que resuelven sus propios problemas.

Lo más importante es que las herramientas están diseñadas para funcionar juntas, como partes de una máquina o bloques de construcción. No sólo se incluyen en el sistema UNIX muchas herramientas y utilidades sino que muchas otras están disponibles como opciones, entre las que se incluyen todas aquellas que se encuentran disponibles como los archivos en Internet sin costo alguno.

Capacidades multiusuario y multitarea

El sistema operativo UNIX puede ser utilizado por computadoras con muchos usuarios o con un solo usuario, ya que es un sistema multiusuario. También es un sistema operativo multitarea, ya que un único usuario puede llevar a cabo más de una tarea al mismo tiempo. Por ejemplo, se puede ejecutar un programa que compruebe la corrección sintáctica de las palabras de un archivo de texto mientras se lee el correo electrónico.

Entorno excelente para redes

El sistema UNIX proporciona un entorno excelente para redes. Ofrece programas y facilidades que proporcionan los servicios necesarios para construir aplicaciones basadas en red, base de la computación distribuida. En los servidores de red, la información y su procesamiento es compartido por diferentes computadoras. El sistema UNIX ha demostrado ser útil en computación cliente/servidor, donde máquinas de una red pueden ser al mismo tiempo clientes y servidores. Este sistema ha sido también básico para el desarrollo de los servicios de Internet y para su propio crecimiento. Consecuentemente, con la importancia de la computación distribuida e Internet está creciendo la popularidad del sistema UNIX.

Portabilidad

El sistema UNIX es mucho más fácil de portar a nuevas máquinas que otros sistemas operativos, es decir se necesita menos trabajo para adaptarlo y ejecutarlo sobre una máquina nueva. La portabilidad del sistema UNIX es consecuencia directa de estar escrito casi completamente en un lenguaje de alto nivel, el lenguaje C; lo que hace posible mover las aplicaciones de un sistema a otro.

Aplicaciones

Se pueden utilizar aplicaciones construidas utilizando órdenes, herramientas y programas del sistema UNIX. Los programas de aplicación llevan a cabo diferentes tipos de tareas, algunas realizan funciones generales que pueden ser utilizadas por una amplia variedad de usuarios del gobierno, la industria y la educación. Éstas se conocen como aplicaciones horizontales e incluyen programas tales como procesadores de texto, compiladores, sistemas de gestión de bases de datos, hojas de cálculo, programas de análisis estadístico y programas de comunicaciones. Otras son específicas de la industria y se conocen como aplicaciones verticales; ejemplos de ellas serían los paquetes de software utilizados para gestionar un hotel, un banco y las terminales operativas de los puntos de venta.

Hay varias clases de aplicaciones que han experimentado un crecimiento explosivo en los últimos años. Una de ellas involucra aplicaciones de red, incluyendo aquellas que permiten a la gente hacer uso de un amplio rango de servicios disponibles en Internet. Destacan entre ellas los exploradores (browsers) para el World Wide Web, tales como Mosaic y Netscape. Otra clase importante de aplicaciones permite a los usuarios crear y visualizar archivos multimedia, incluyendo sonido, imágenes y video. La videoconferencia de escritorio es un ejemplo de una aplicación que combina conexión en red y multimedia; muchos fabricantes ofrecen actualmente esta aplicación sobre sus plataformas UNIX.

Utilidades

El sistema UNIX contiene varios cientos de utilidades o programas de usuario. Las órdenes también se conocen como herramientas, pues pueden utilizarse independientemente o en forma conjunta de diversas maneras para llevar a cabo tareas útiles. Estas utilidades se pueden ejecutar invocándolas por su nombre a través del shell, es por ello por lo que se conocen como órdenes.

Una de las principales diferencias entre el sistema UNIX y los sistemas operativos anteriores es la facilidad con que los nuevos programas pueden ser instalados; el shell sólo necesita conocer dónde buscar las órdenes, y esto es definible por el usuario.

Se pueden realizar múltiples tareas empleando las utilidades estándar suministradas con el sistema UNIX. Hay utilidades para edición y procesamiento de texto, gestión de información, comunicaciones electrónicas y redes, para realizar cálculos, desarrollar programas de computadoras, y para la administración del sistema entre otros muchos propósitos. Una de las características del sistema es la disponibilidad de una amplia variedad de utilidades opcionales, libres de gastos o bien mediante la adquisición a través de los vendedores de software.

2.3 Nacimiento del sistema operativo UNIX

“La historia del sistema UNIX data de los años sesenta, cuando los Laboratorios Bell de AT&T y el fabricante de computadoras GE (General Electric) trabajaron sobre un sistema operativo experimental denominado MULTICS (Multiplexed Information and Computing System) — Información Multiplexada y sistema de computación—, que fue diseñado como sistema operativo interactivo para la computadora GE 645, permitiendo la compartición de información al tiempo que proporcionaba seguridad. El desarrollo sufrió muchos retrasos y las versiones de producción resultaron lentas y con grandes necesidades de memoria. Por una serie de razones, los Laboratorios Bell abandonaron el proyecto. Sin embargo, el sistema MULTICS implementó muchas características innovadoras y produjo un entorno de computación excelente”⁹.

⁹ Rosen, Kenneth, et al. *UNIX. Sistema V Versión 4*. p. 10

En 1969, Ken Thompson, uno de los investigadores de los Laboratorios Bell involucrado en el proyecto MULTICS, Dennis Ritchie, Rudd Canaday y sus colegas crearon un sistema operativo multitarea, incluyendo un sistema de archivos, un intérprete de órdenes y algunas utilidades para una nueva máquina. Puesto que el nuevo sistema operativo multitarea podía soportar dos usuarios simultáneamente, se le llamó UNICS (Uniplexed Information and Computing System) — Información Uniplexada y Sistema de Computación—; el primer uso de este nombre se atribuye a Brian Kernigham. El nombre se cambió ligeramente a UNIX en 1970 y ha permanecido así desde entonces.

En 1973 Ritchie y Thompson volvieron a escribir el núcleo en el lenguaje de programación C, un lenguaje de alto nivel a diferencia de la mayor parte de los sistemas escritos para máquinas pequeñas, que utilizaban generalmente un lenguaje ensamblador. La escritura del sistema operativo en C hacía más fácil su mantenimiento y portabilidad a otras máquinas. La popularidad del sistema UNIX creció debido a sus innovaciones y a que estaba escrito compactamente en un lenguaje de alto nivel con código que podía modificarse de acuerdo a las preferencias individuales. AT&T no ofreció comercialmente el sistema UNIX porque en ese tiempo no estaba en el negocio de las computadoras. Sin embargo, sí permitió la disponibilidad de este sistema a universidades, firmas comerciales y al gobierno por un costo simbólico.

Hacia 1974 comenzó a utilizarse ampliamente en los Laboratorios Bell la cuarta edición del sistema UNIX. —Las versiones del sistema UNIX producidas por los grupos de investigación de los Laboratorios Bell se han conocido tradicionalmente como ediciones—. Hacia 1977 surgieron la quinta y sexta ediciones; éstas contenían bastantes más herramientas y utilidades. El número de máquinas que ejecutaba el sistema UNIX, fundamentalmente en los Laboratorios Bell y en las universidades se incrementó en más de 600 en 1978. La séptima edición, el ascendente directo del sistema operativo UNIX, disponible hoy día, salió en 1979.

El sistema III de UNIX, basado en la séptima edición, se convirtió en 1982 en la primera versión comercial del sistema UNIX de AT&T. Sin embargo, una vez lanzado el sistema III, AT&T, a través de su subsidiaria Western Electric, continuó vendiendo versiones del sistema UNIX. El sistema III de UNIX, las diferentes ediciones de investigación y las versiones experimentales se distribuyeron a las Universidades y otros laboratorios de investigación. Con frecuencia resultaba imposible para los informáticos saber si una característica particular formaba parte de la estructura principal del sistema UNIX o era sólo parte de una de sus versiones.

Para eliminar esta confusión sobre las variedades del sistema UNIX, AT&T introdujo el UNIX Sistema V Versión 1 en 1983. (El Sistema IV de UNIX sólo existió como versión interna AT&T). Con este sistema AT&T prometió por primera vez mantener compatibilidad ascendente en sus futuras versiones del sistema UNIX. Esto significaba que los programas construidos sobre la Versión 1 podrían continuar operando con futuras versiones del sistema V.

AT&T ofreció el UNIX Sistema V Versión 2 en 1985. La Versión 2 introdujo protección de archivos durante cortes de energía, bloqueo de archivos y registros para uso exclusivo por un programa, características de control de trabajo y administración ampliada del sistema. La Versión 2.1 introdujo dos características adicionales de interés para los programadores: la paginación por demanda, que permite ejecutar procesos que requieren más memoria de la que se dispone físicamente, y bloqueo de archivos y registros.

En 1987, AT&T introdujo el UNIX Sistema V Versión 3.0; ésta incluía un enfoque de redes simple y consistente. Estas capacidades incluyen streams, utilizado para construir software de redes, el sistema de archivos remoto, utilizado para compartir archivos a través de redes, y la interfaz a Nivel de Transporte (TLI), utilizada para construir aplicaciones que utilizan redes. La Versión 3.1 hizo al Sistema V de UNIX adaptable internacionalmente, soportando conjuntos de caracteres y formatos de hora y fecha más amplios. También proporciona los diferentes aumentos de rendimiento en el uso de la memoria y en las copias de seguridad y la recuperación de archivos. La versión 3.2 proporcionó seguridad ampliada del sistema, incluyendo la visualización de la hora de presentación última del usuario, el registro de los intentos de inicialización sin éxito y un archivo oculto de contraseñas que impide a los usuarios la lectura de contraseñas cifradas. La Versión 3.2 también introdujo el Framed Access Command Environment (FACE), que proporciona una interfaz de usuario orientada a menú. La Versión 4 unifica varias versiones de sistema UNIX que han sido desarrolladas dentro y fuera de AT&T.

2.4 Factores que influyeron en el desarrollo de UNIX

“Muchas innovaciones importantes del sistema UNIX se han desarrollado en la Universidad de California, Berkeley y en el Departamento Defense’s Advanced Research Projects Agency (DARPA) de los Estados Unidos [Fig. 2.2] Algunas de estas ampliaciones han formado parte del Sistema V de UNIX en las primeras versiones y muchas más se han introducido en la versión 4”¹⁰.

La U. C. Berkeley se involucró en el sistema UNIX en 1974, comenzando con la cuarta edición. El desarrollo de la versión de Berkeley del sistema UNIX fue promovida por Ken Thompson en 1975 en el departamento de informática. En Berkeley, Ken transportó la sexta edición a una máquina, haciendo que el sistema estuviese disponible para un gran número de usuarios. Los estudiantes graduados Bill Joy y Chuck Haley hicieron buena parte del trabajo de la versión Berkeley. Ellos diseñaron un editor denominado ex y produjeron un compilador de Pascal. Joy diseñó un paquete que llamó el Berkeley Software Distribution (BSD). También hizo muchas otras innovaciones valiosas, incluyendo el shell C y el editor orientado a pantalla vi —una ampliación de ex—. En 1978 se hizo la Segunda Distribución de Software de Berkeley; conocida como 2BSD. En 1979 se distribuyó 3BSD, basada en la 2BSD y la séptima edición, proporcionando memoria virtual para permitir que pudiesen ejecutarse grandes programas.

A finales de los setenta, el Departamento Defense’s Advanced Research Projects Agency (DARPA) de los Estados Unidos decidió basar su entorno de computación universal en el sistema UNIX. DARPA decidió que el desarrollo de su versión del sistema UNIX se llevase a cabo en Berkeley, consolidando la 4BSD.

En 1983 apareció la 4.1BSD, que disponía de mayor rendimiento. La 4.2BSD también apareció en el mismo año, introduciendo características de red, incluyendo redes TCP/IP, que pueden ser utilizadas para transferir archivos y presentaciones remotas y un nuevo sistema de archivos que acelera el acceso a los mismos. La Versión 4.3BSD apareció en 1987, con pequeños cambios sobre la 4.2BSD.

¹⁰ Ídem. p.13

El sistema BSD ha continuado su evolución pero, con la incorporación de la mayor parte de sus capacidades más importantes al UNIX Sistema V Versión 4, no tiene la influencia tan grande que tuvo durante la evolución de UNIX. La 4.4BSD es la última versión de BSD, que incluye una gran variedad de mejoras, muchas de las cuales se refieren a capacidades de red.

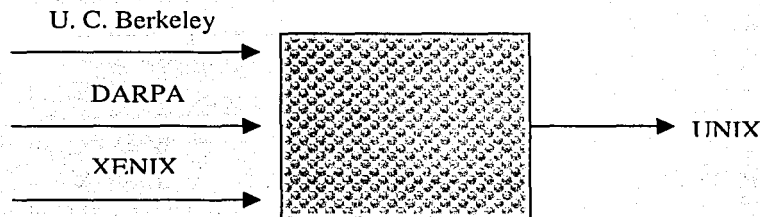


Fig. 2.2 Factores que influyeron en el desarrollo de UNIX.

El sistema XENIX

En 1980, Microsoft introdujo el sistema XENIX, una variante del sistema UNIX, diseñado para ejecutarse sobre microcomputadoras. La introducción del sistema XENIX llevó las capacidades del sistema UNIX a máquinas de sobremesa, capacidades que anteriormente sólo estaban disponibles en grandes computadoras. El sistema XENIX se basó originalmente en la séptima edición, con algunas utilidades presentadas de la 4.1BSD;0 en la Versión 3.0 del sistema XENIX, Microsoft incorporó nuevas características del Sistema III del UNIX de AT&T y en 1985 el sistema XENIX se desplazó hacia uno basado en el Sistema V de UNIX.

En 1987 XENIX fue portado por la Operación Santa Cruz (SCO) a máquinas basadas en el 80386, una compañía que ha trabajado con Microsoft sobre el desarrollo de XENIX. En 1987, Microsoft y AT&T comenzaron a desarrollar juntos la fusión de XENIX y el Sistema V de UNIX consumándose en el UNIX Sistema V Versión 3.2. Este esfuerzo logró una versión unificada del sistema UNIX que puede ejecutarse en sistemas que van desde las computadoras personales de sobremesa hasta las supercomputadoras. De todas las variantes del sistema UNIX, el sistema XENIX ha conseguido el mayor número de máquinas instaladas. XENIX, aunque todavía soportado por SCO, ya no supone un factor importante en el mundo de UNIX.

2.5 UNIX Sistema V Versión 4

“UNIX Sistema V Versión 4 fue desarrollado en los Laboratorios Bell (con contribuciones de Sun Microsystems) en la década de los 80. Se combinaron las mejores variantes de UNIX de aquella época —UNIX Sistema V de AT&T, XENIX de Microsoft y SunOS de Sun Microsystems— en un solo sistema operativo. Durante los años ochenta, muchas empresas de computadoras fueron conscientes de la ventaja competitiva que tenía AT&T y Sun Microsystems debido a su control sobre UNIX Sistema V. Como consecuencia de esta situación, algunos grupos de vendedores de computadoras formaron alianzas, bien para unirse a la evolución de UNIX Sistema V o para proteger sus intereses comerciales de aquellos otros vendedores asentados en el mercado”¹¹.

Estándares

El uso de diferentes versiones del sistema UNIX ocasionaba problemas a las personas encargadas de desarrollar aplicaciones para un amplio rango de computadoras que soportaban el sistema operativo UNIX. Para resolver este problema se han desarrollado varios estándares, los cuales definen las características que deberá tener un sistema para que puedan construirse aplicaciones que funcionen sobre cualquier sistema que se adecúe al estándar. Uno de los objetivos de la Versión 4 es unificar las variantes importantes del sistema UNIX en un único producto estándar.

La definición de la interfaz del Sistema V

Para que el Sistema V de UNIX se convierta en un estándar de la industria, otros vendedores necesitan poder probar sus versiones para acomodarlas a la funcionalidad del Sistema V. En 1983, AT&T publicó la Definición de la Interfaz del Sistema V (SVDI), que especifica cómo debe comportarse un sistema operativo para cumplir con el estándar.

Las personas que desarrollan aplicaciones pueden construir programas con garantía de que funcionen sobre cualquier máquina que opere con una versión del sistema UNIX conforme a SVDI. Además, la SVDI especifica características del sistema UNIX de las que existen garantías de no cambiar en futuras versiones, de manera que existe la garantía de que las aplicaciones puedan ejecutarse sobre todas las versiones del Sistema V de UNIX. Los vendedores pueden comprobar si sus versiones del Sistema UNIX se ajustan a SVDI ejecutando el System V Verification Suite desarrollado por AT&T. Naturalmente, el UNIX Sistema V Versión 4 se ajusta a SVDI.

¹¹ Ídem. p.23

TESIS CON
FALLA DE ORIGEN

POSIX

“Un esfuerzo independiente para definir un estándar de entorno de sistema operativo comenzó en 1981 por iniciativa de /usr/group, una organización constituida por usuarios del sistema UNIX que querían asegurar la portabilidad de las aplicaciones. Publicaron un estándar en 1984. Debido a la magnitud del trabajo, en 1985 el comité de trabajo sobre estándares se fusionó con el proyecto 1003 (P1003) del Institute for Electrical and Electronics Engineers (IEEE). El objetivo del P1003 fue establecer un conjunto de estándares ANSI (American National Standards Institute). Los estándares que los diferentes grupos de trabajo del P1003 están estableciendo se conocen como *Portable Operating System Interface for Computer Environment (POSIX)*.”¹²

POSIX será una familia de estándares que define cualquier aplicación que interactúa con un sistema operativo. Entre las áreas cubiertas por los estándares POSIX están las llamadas al sistema, bibliotecas, herramientas, interfases, verificación y prueba, características en tiempo real y seguridad. El estándar POSIX que ha recibido mayor atención es el P1003.1 (conocido como POSIX.1), que define la interfaz del sistema. Otro estándar importante es el P1003.2 (POSIX.2) que versa sobre shells y utilidades. El POSIX 1003.3 cubre los métodos de prueba sobre el cumplimiento de los estándares POSIX; el estándar POSIX 1003.4 cubre las extensiones sobre tiempo-real. Existen más de 20 estándares diferentes que han sido desarrollados o están en vías de desarrollo.

POSIX ha sido confirmado por el National Institute of Standards and Technology (NIST), como parte del Federal Information-Processing Standard (FIPS), que debe ser cumplido por todas las computadoras adquiridas por el gobierno federal de los Estados Unidos.

X/OPEN

Otro conjunto importante de estándares del sistema UNIX lo proporciona X/Open, un consorcio internacional de vendedores de computadoras establecido en 1984. X/Open adopta los estándares e interfaces existentes pero no desarrolla sus propios estándares. X/Open comenzó con vendedores europeos de computadoras pero hoy engloba muchas compañías de Estados Unidos, entre las que se encuentran AT&T, DEC, Hewlett-Packard, IBM, Sun, Unisys, Bull, Ericson, Olivetti y Phillips. El objetivo de X/Open consiste en estandarizar las interfaces software.

Uno de los mayores problemas en la industria del UNIX (y de los sistemas abiertos) es que los vendedores de software deben dedicar un gran esfuerzo para portar un determinado software a diferentes sistemas UNIX. En 1993, para mitigar este problema, la X/Open asumió la responsabilidad de controlar la evolución de una especificación común de interfaz de programación de aplicaciones (API). Esta aplicación permitirá a un vendedor de software sobre sistema UNIX que cumpla esta especificación.

¹² Ídem. p.16

2.6 Características de la Versión 4

El conjunto unificado de órdenes

El conjunto unificado de órdenes del UNIX Sistema V Versión 4 se construyó fusionando el conjunto de órdenes del UNIX Sistema V Versión 3.2 con las órdenes más populares de los sistemas BSD, XENIX y SunOS, al tiempo que se añadían nuevas órdenes y se actualizaban algunas antiguas.

Shells

La Versión 4 soporta cuatro intérpretes de órdenes, o shells. Como el shell por omisión, que dispone de una versión ampliada del shell del Sistema V de UNIX estándar. Además del shell estándar, la Versión 4 incluye otros tres: el shell job, el shell Korn y el shell C. El primero incorpora características de control al shell estándar del Sistema V; el segundo ofrece un superconjunto de las características del shell del Sistema V, incluyendo la historia de órdenes, la edición de la línea de orden y características ampliadas de programación; y, el tercero procedente del sistema BSD, proporciona capacidad de control de trabajos, historia de órdenes, capacidades de edición y otras características interactivas. Algunos usuarios prefieren utilizar otros shells distintos de los que ofrece el UNIX Sistema V disponibles desde Internet. Uno de los aspectos más atractivos del Sistema UNIX es que tales cosas son posibles.

El árbol de directorios

En el sistema UNIX los archivos se disponen en directorios jerárquicos. La disposición del árbol de directorios del sistema UNIX se ha cambiado en la Versión 4 para acomodar los entornos de red y la compartición de archivos remotos. Los archivos del árbol de directorios se dividen en:

- Archivos necesarios para la carga del sistema.
- Archivos compatibles que permanecen estáticos a lo largo de la vida del sistema.
- Archivos de usuario.
- Archivos y directorios del sistema que cambian a lo largo de la vida del sistema.

Entrada/salida y acceso al sistema

La Versión 4 incluye ampliaciones que hacen más fácil a los programadores utilizar nuevos dispositivos. Esto se lleva a cabo utilizando el Device-Kernel Interface (DKI) entre el núcleo y el software del dispositivo.

La Versión 4 introduce una interfaz común para acceso al sistema. Este acceso puede ser a través de un terminal, a través de una red de área local o por acceso remoto.

Procesamiento en tiempo real

Procesamiento en tiempo real es la ejecución de programas de computadora en intervalos de tiempo especificados sin retardo. Es diferente al procesamiento en tiempo compartido, que puede retardar la ejecución de los procesos y que no garantiza la ejecución en un tiempo especificado.

El UNIX Sistema V Versión 4 incluye nuevas características para soportar procesamiento en tiempo real. Versiones anteriores de UNIX Sistema V sólo ofrecían un entorno de propósito general de tiempo compartido que planificaba los procesos para atender las diferentes necesidades de los usuarios. Con este tipo de planificación los procesos reciben pequeños intervalos de tiempo de procesador, separándolos por intervalos utilizados por otros procesos y no tiene control sobre su prioridad con respecto a los otros procesos. El procesamiento en tiempo real requiere la dedicación de un intervalo de tiempo a un proceso particular y la posibilidad de que un proceso controle su planificación.

Para acomodar la planificación en tiempo real, la Versión 4 soporta dos clases diferentes de procesos: de tiempo compartido y de tiempo real. Para gestionar los procesos de tiempo real se utilizan nuevas llamadas al sistema, incluyendo varias adaptadas del sistema BSD.

Administración del sistema

La Versión 4 incluye nuevas características para hacer más fácil la administración y el mantenimiento. Se ha simplificado la interfaz de menú para la administración del sistema. Se han hecho mejoras sobre los procedimientos de seguridad y restauración del sistema. Se ha simplificado la instalación del software.

Conexión en red

Las redes son una de las áreas claves de las ampliaciones de la Versión 4. Se han incorporado muchas capacidades de red de los sistemas BSD y SunOS. Las capacidades de red del sistema BSD incluyen al paquete Internet TCP/IP ampliamente utilizado, que a su vez incluye órdenes para transferencia de archivos, presentación remota y ejecución remota. También la Versión 4 incluye la interfaz de red de conexiones lógicas de red (sockets), que se utilizan para construir aplicaciones basadas en red. Del SunOS, la Versión 4 ha incorporado características de red que incluyen al sistema de archivos de red (NFS) para compartición de archivos remotos, el protocolo estándar de industria Remote Procedure Call (RPC), para que un procedimiento sobre una computadora llame a un procedimiento sobre una computadora remota, y el External Data Representation (XDR), que especifica un formato para los datos permitiendo su intercambio entre sistemas que incluso tengan arquitectura de hardware, sistema operativo y/o lenguaje de programación.

Las nuevas características de red de la Versión 4 incluyen la característica Network Selection, que permite a una aplicación seleccionar una red sobre la que comunicar, un mecanismo de translación nombre-a-dirección utilizado para que las máquinas cliente determinen la dirección de los servidores que proporcionan servicios particulares, y uucp independiente del medio, que permite la transferencia de archivos sobre cualquier clase de red.

Interfaces de usuario

Quizá la diferencia más visible respecto a las versiones anteriores es que la Versión 4 ofrece interfaces de usuario estándar y sistema de ventanas para aplicaciones basadas en carácter y gráficas. La interfaz gráfica de usuario incluida con la Versión 4 se denomina OPEN LOOK, que ofrece una forma consistente, efectiva y eficaz de interactuar con las aplicaciones. Proporciona varias posibilidades de gestión y operación de ventanas, utilización de barra de desplazamiento, clavijas de pulsar, pulsadores, menús, ventanas emergentes y facilidades de ayuda. Con esta interfaz los usuarios pueden aprender rápidamente las aplicaciones OPEN LOOK. Otra interfaz gráfica de usuario importante, MOTIF, desarrollada originalmente por la Open Software Foundation, ha llegado a ser mucho más popular que OPEN LOOK en los últimos años; MOTIF está incluido en algunas versiones del sistema UNIX.

Desde que surgió el Sistema V Versión 4 se ha desarrollado una interfaz gráfica de usuario estándar conocida como Common Desktop Environment (CDE). El CDE ha sido adoptado por muchos de los desarrolladores importantes de software del sistema UNIX.

La Versión 4 también proporciona el Framed Access Command Environment (FACE), que es una interfaz para terminales de usuario basada en carácter. FACE presenta el entorno del sistema UNIX a través de ventanas que contienen menús.

2.7 Variantes del sistema operativo UNIX

“UNIX Sistema V Versión 4 ha obtenido un gran impacto en el mercado de los sistemas UNIX. Casi todos los vendedores principales ofrecen versiones de UNIX que se basan en UNIX Sistema V Versión 4 y que se ajustan a los estándares para sistemas abiertos basados en él y todas las versiones de UNIX comparten muchas características con la versión 4”¹³.

Aunque existen decenas de variantes de UNIX, la mayoría comparten una gran cantidad de características, pudiéndose portar software entre la mayoría de variantes modernas casi directamente.

UNIXWARE

UnixWare ha sido la denominación utilizada por Novell para sus productos basados en UNIX Sistema V; estos productos los ofrece ahora Santa Cruz Operation a raíz de la venta de dichos productos por Novell en septiembre de 1995; la última versión de UnixWare es la 2, basada en la integración de UNIX Sistema V Versión 4.2 y Novell NetWare, soportado por la computación cliente/servidor.

UnixWare cumple, entre otros, con los estándares POSIX. Existen dos versiones de UnixWare 2, la versión Personal Edition, diseñada para su uso como sobremesa y cliente servidor, y UnixWare 2 Application Server, diseñada para usar en servidores.

UnixWare sólo está disponible en la actualidad en computadoras basadas en procesadores Intel y será portada a computadoras basadas en arquitectura RISC, como las máquinas que se basan en procesadores SPARC. Se han escrito más de 3000 aplicaciones para UnixWare, soportando la interfaz gráfica de usuario MOTIF.

UNIX SCO

Santa Cruz Operation (SCO) basa sus sistemas operativos en UNIX Sistema V Versión 3.2, una versión de UNIX Sistema V Versión 3 diseñada para usar sobre procesadores Intel 80386, los dos entornos operativos principales de SCO son Open Desktop 5.0 y Open Server 5.0. Open Desktop 5.0.

Está diseñado para ser usado en computadoras personales basadas en Intel, ejecutando la mayoría de aplicaciones DOS, Windows y XENIX; SCO Open Server incluye un amplio rango de capacidades de red, entre otros TCP/IP, NFS y soporte para varios protocolos LAN.

¹³ Ídem. p.25

IRIX

IRIX es la versión propietaria de UNIX Sistema V Versión 4 que proporciona Silicon Graphics para usar sobre estaciones de trabajo basadas en MIPS. IRIX es un sistema operativo a 64-bits, que es una de las características que optimizan su rendimiento para aplicaciones gráficas que requieren un procesamiento intensivo de la CPU. La versión actual de IRIX es IRIX 5.3 incorporando lo esencial de UNIX sistema V Versión 4 y 4.2.

HP-UX

La variante del sistema operativo UNIX desarrollado por Hewlett-Packard para usar en sus computadoras y estaciones de trabajo se denomina HP-UX, basado en UNIX Sistema V Versión 2.0, con diversas ampliaciones, ha sido diseñado conforme a muchos de los estándares para UNIX.

AIX

La versión de IBM del sistema operativo UNIX se denomina AIX, desarrollada inicialmente para ser utilizada sobre estaciones de trabajo IBM. AIX Versión 4 es la última versión de AIX; esta versión se ejecuta sobre procesadores POWER y PowerPC. Se basa en UNIX Sistema V Versión 3 y tiene características del BSD4.3; incluye soporte MOTIF y una implementación parcial del Common Desktop Environment. Existen más de 10.000 aplicaciones diferentes que se ejecutan sobre plataformas AIX.

LINUX

La variante Linux del sistema operativo UNIX fue diseñada originalmente para ser ejecutada en computadoras personales basadas en procesadores Intel 80x80. ha sido y sigue siendo portada a varias plataformas. El desarrollo de Linux lo inició Linus Torvalds en 1991. Al contrario de lo que ocurre con otras versiones de UNIX, Linux es libre de uso. No obstante, posee un copyright bajo los términos de licencia pública general, para que pueda ser vendido y distribuirlo sin permitir al comprador que lo copie libremente. Linux cumple con el estándar POSIX.1.

A/UX

A/UX (de Apple's UNIX) es una implementación del sistema operativo UNIX de Apple. A/UX 3 funde la funcionalidad del sistema UNIX con el sistema operativo Macintosh System 7. Se basa en UNIX Sistema V Versión 2.2, pero posee muchas extensiones de la interfaz del Sistema V y el estándar POSIX. Dado que A/UX 3 incorpora el sistema operativo System 7 para el Macintosh, casi todas sus aplicaciones pueden usarse bajo AU/X, así como portar aplicaciones UNIX al entorno.

2.8 Estructura del sistema operativo UNIX

El sistema operativo UNIX está conformado por varios componentes. Entre los que se encuentran el núcleo, el shell, el sistema de archivos y las órdenes (o programas de usuario). Las relaciones entre el usuario, el shell, el núcleo y el hardware subyacente se representa en la figura siguiente.

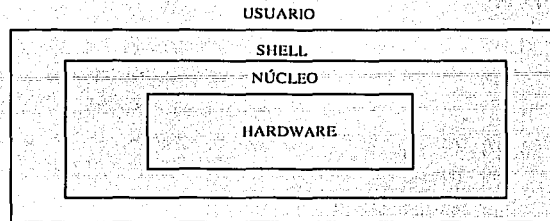


Fig. 2.3 Estructura del sistema operativo UNIX.

Organización de la información

El sistema operativo UNIX está diseñado para manejar información contenida normalmente en discos. Para que esta manipulación sea realmente efectiva, es necesario que la información esté organizada de alguna forma. La manera estándar de organizar la información es en archivos. Los archivos son localizados dentro del disco porque son apuntados desde un lugar determinado, al que se le denomina directorio.

Sin embargo en UNIX no se utiliza un único directorio para apuntar a todos los archivos del sistema, sino que se crea una estructura jerárquica de directorios conocida como estructura en árbol. Aunque en un principio pueda parecer una estructura excesivamente complicada por la amplia ramificación de los directorios, esta organización permite agrupar los archivos de los diferentes usuarios e incluso de las diferentes aplicaciones en directorios separados, con lo que se evita que interfieran entre sí. Por otro lado, un mismo usuario puede organizar su propia información separando los archivos en diferentes directorios de acuerdo a su contenido como se muestra en la Fig. 2.4.

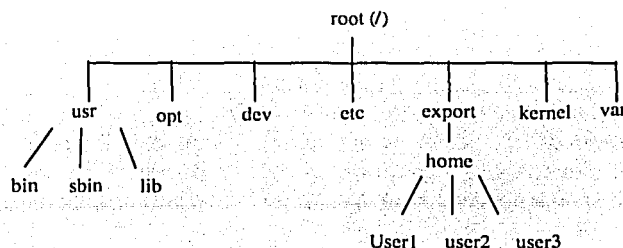


Fig. 2.4 Estructura del sistema de archivos.

Concepto de archivo

La estructura fundamental que utiliza el UNIX para almacenar información es el archivo; mantiene la pista de cada uno de los archivos asignándoles a cada uno, un número de identificación (i-number) que apunta a la tabla de i-nodos. El usuario, no obstante, indicará cada archivo por medio de su nombre asignado por el propio usuario. El UNIX mantiene toda la información propia del archivo, excepto su nombre, en su inodo correspondiente. El nombre se encuentra en la entrada correspondiente del directorio al que pertenece.

Todos los archivos UNIX son tratados como una simple secuencia de bytes (caracteres), comenzando en el primer byte del archivo y terminando con el último. Un byte en concreto dentro del archivo, es identificado por la posición relativa que ocupe en el archivo. La organización de los archivos en registros o bloques de una longitud fija, corresponde a la forma tradicional de organización de los datos en el disco. El UNIX protege específicamente al usuario del concepto de registros o bloques, en su lugar, puede permitir que el usuario divida al archivo en registros, de acuerdo a un byte o conjunto de bytes especiales. Los programas son libres para organizar sus archivos, con independencia de la forma en que los datos estén almacenados en el disco.

Concepto de directorio

Prácticamente para todos los efectos, un directorio se comporta como un archivo, con la característica de que sus registros son de longitud fija. A pesar de lo comentado anteriormente, el contenido de los directorios no apunta directamente a los bloques de datos de los archivos o subdirectorios que dependen de él, sino a unas tablas (i-nodos) separadas de la estructura. Es desde estas tablas desde las que se apunta a los bloques físicos de los archivos dentro del disco como se muestra en la figura 2.5.

Cada registro en el directorio tiene una longitud de 16 bytes de los que:

- Los dos primeros contienen un apuntador (en binario) a la tabla de los i-nodos (i-number).
- Los catorce restantes contienen el nombre del archivo o subdirectorio (el número bytes por registro depende de la versión del UNIX, ya que hay versiones en que la longitud del nombre de un archivo puede llegar a 255 bytes).

Cualquier directorio contiene un mínimo de dos entradas:

- Una referencia a sí mismo (.).
- Una referencia al directorio de que dependo o directorio padre (..), salvo en el directorio raíz que, al ser el comienzo de la estructura, esta referencia es también sobre sí mismo.

i-NODO	Nombre del archivo
20	.
2	..
45	datos
53	work

Fig. 2.5 Estructura de un directorio.

El sistema de archivos

El conjunto formado por la estructura de archivos y directorios y las tablas de ínodos, se denomina Sistema de Archivos o File System. Cada sistema de archivos está residente en un soporte físico, normalmente disco magnético, pero si el disco es de gran capacidad puede dividirse en varios discos lógicos o "particiones", cada uno de los cuales sería un sistema de archivos independiente. En el caso de que el sistema disponga de más de un disco físico, cada uno será forzosamente un file system. Por lo tanto se puede asociar el concepto de sistema de archivos al de disco lógico o partición de disco.

Es importante no perder la idea de que cada sistema de archivos dispondrá de su propia tabla de í-nodos que apuntarán a los bloques de datos de los archivos y directorios contenidos en él.

El sistema de archivos que contiene el directorio raíz (normalmente el primer disco o la primera partición del primer disco en su caso) es el principal. En principio, cuando se carga el sistema operativo, la única parte accesible de la estructura es la contenida en el file system principal. Para poder acceder a la información contenida en los restantes sistemas de operación es necesario realizar una operación denominada "montaje". Esta operación consiste en enlazar cada sistema de archivos con directorios vacíos de la estructura principal. A partir de ese momento, los archivos y directorios de los sistemas de archivos secundarios figurarán dentro de la estructura, colgando del directorio de montaje.

Una vez montados todos los sistemas de archivos, la estructura es única, por lo que el disco o la partición en que esté físicamente un determinado archivo, es transparente para el usuario. La función de montaje y desmontaje de los sistemas de archivo es, normalmente, realizada por el administrador del sistema.

El sistema de archivos, o file system, de UNIX está caracterizado por:

- Una estructura jerárquica,
- Tratamiento consistente de los datos,
- La habilidad de crear y borrar archivos,
- Crecimiento dinámico de archivos,
- La protección de los datos,
- El tratamiento de los dispositivos periféricos como archivos.

Cada sistema de archivos consta fundamentalmente de las siguientes partes: (Fig. 2.6)

❖ Bloque de arranque:

Este bloque, que es el primero de cada sistema (bloque cero), está reservado para un programa de carga. El bloque cero no tiene ningún significado en el sistema de archivos. Toda la información del sistema comienza en el bloque uno del dispositivo. Sólo se utiliza en el sistema de archivos principal.

❖ Súper bloque:

El súper bloque es el bloque uno del dispositivo. Este bloque contiene información sobre el sistema de archivos, tales como su tamaño en bloques, el nombre del sistema de archivos, número de bloques reservados para i-nodos, la lista de i-nodos libres y el comienzo de la cadena de bloques libres. También contiene el nombre del volúmen, momento de la última actualización y tiempo del último backup. Siempre reside en un bloque de 512 bytes.

❖ I-nodos:

A continuación del súper bloque están situados los bloques que contienen los i-nodos. El número de bloques de i-nodos varía dependiendo del número de bloques del sistema de archivos. El número de i-nodos está especificado en el súper bloque. Es una tabla que contiene información sobre las características de los archivos. Esencialmente es el bloque de control de los archivos. Hay un i-nodo por cada directorio y archivo del sistema de archivos.

El i-nodo contiene una descripción del directorio o archivo, así como el lugar físico que ocupan sus bloques de datos. Los i-nodos sólo apuntan a los archivos o directorios de su mismo sistema de archivos.

❖ Bloques de datos:

El resto del espacio del dispositivo lógico consta de bloques de datos que contienen los datos actualmente almacenados en los archivos. Algunos bloques de datos sirven como bloques indirectos, conteniendo números de bloques (direcciones) de grandes archivos.

Bloque de arranque	Super-bloque	Lista de inodos	Bloques de datos
--------------------	--------------	-----------------	------------------

Fig. 2.6 Estructura de un sistema de archivos.

El sistema de archivos está organizado en una estructura jerárquica. Este sistema jerárquico permite agrupar la información de los usuarios de una forma lógica, así como manipular eficientemente un grupo de archivos como una sola unidad. No tiene ninguna limitación en su desarrollo, permitiendo un crecimiento dinámico tanto en su número como en su tamaño. El UNIX trata a los archivos de entrada o de salida del mismo modo que cualquier otro archivo. Los usuarios deberán "caminar" a través del árbol hasta encontrar el archivo deseado. Dentro del UNIX el usuario puede acceder a cuatro clases de archivos:

- Directorios.
- Ordinarios.
- Especiales.
- Fifo (Pipes).

El shell

El shell es la parte visible por el usuario del UNIX. Es el intermediario entre el sistema operativo básico (kernel) y el usuario. Existen varios tipos de shell:

- El considerado estándar es el Bourne Shell, distribuido por AT&T con el UNIX. La versión original fue desarrollada por Stephen Bourne en los Laboratorios Bell.
- Otro de los más difundidos es el C Shell, desarrollado en la Universidad de Berkeley en su mayor parte por Bill Joy. Fue diseñado teniendo en cuenta que los usuarios serían programadores en lenguaje C.
- Otra versión ampliamente difundida es el Korn Shell, interfaz escrita por David Korn.

Existe en UNIX un conjunto de variables denominadas variables de entorno. Dichas variables son creadas habitualmente por el propio shell en concordancia con el modo y las características dadas por el administrador del sistema al dar de alta al usuario. Estas variables son las que definen el propio entorno de trabajo y son redefinibles por el usuario, lo que agrega flexibilidad al sistema y permite adaptar el entorno al gusto o necesidad de cada uno.

Cada vez que se introduce un comando, el shell analiza la línea, verifica la sintaxis y lo ejecuta. El ciclo se repite hasta que el usuario se desconecta. Entonces, el programa shell termina su ejecución.

El shell es, por tanto, un procesador de comandos. Pero también ejerce otras responsabilidades: se encarga de la redirección de entradas y salidas, es también un lenguaje de programación interpretado, es el encargado de proveer los comandos y variables para controlar y modificar el entorno de usuario (es decir, tiene la función de ser interfaz de usuario).

El núcleo

El núcleo es la parte del sistema operativo que interactúa directamente con el hardware de una computadora, a través de los controladores de dispositivo que están incorporados en el núcleo. Proporciona conjuntos de servicios que pueden ser utilizados por programas, aislando a estos programas del hardware subyacente. Las funciones principales del núcleo son la gestión de la memoria, el control de acceso a la computadora, el mantenimiento del sistema de archivos, el manejo de las interrupciones (señales que finalizan la ejecución), el manejo de errores, la realización de los servicios de entrada y salida (que permiten a las computadoras interactuar con terminales, dispositivos de almacenamiento e impresoras) y la asignación de recursos de la computadora (tales como el CPU o dispositivos de entrada/salida) entre los usuarios.

Los programas interactúan con el núcleo a través de 100 llamadas al sistema aproximadamente. Las llamadas al sistema dicen al núcleo que lleve a cabo diferentes tareas para el programa, tales como abrir un archivo, escribir u obtener información sobre él, ejecutar un programa, terminar un proceso, etc.

2.9 Sistema Operativo Solaris

El sistema operativo original de Sun Microsystems fue denominado SunOS. Estaba basado en UNIX Sistema V Versión 2 y BSD 4.3. En 1991, Sun Microsystems crea SunSoft como empresa subsidiaria para desarrollar y distribuir software, incluyendo sistemas operativos. En un inicio, SunSoft comenzó la tarea de migrar el sistema operativo SunOS a una nueva versión de UNIX basada en UNIX Sistema V Versión 4. La primera versión de SunSoft de UNIX, Solaris 1.0 (fig. 2.7) fue una versión mejorada de SunOS. Con Solaris 2.0, SunSoft se movió hacia un sistema operativo basado en SVR4. SunSoft tiene una gran base de instalaciones UNIX por encima del millón y medio de máquinas. La versión actual de Solaris es Solaris 2.7. Actualmente existen versiones de Solaris 2.7 para procesadores Intel, SPARC y PowerPC. Existen versiones independientes de servidor y sobremesa de Solaris. Hay disponibles un amplio rango de herramientas de desarrollo para Solaris 2.7. Solaris incluye mayor soporte para la interfaz gráfica de usuario.

SunSoft se encuentra activo en los foros importantes de estándares UNIX. Solaris 2.7 cumple con POSIX.1 y POSIX.2. Más de 9.000 aplicaciones se ejecutan sobre plataformas Solaris.

Solaris es un entorno operativo vendido por SunSoft, la división de software de Sun Microsystems. En el corazón de Solaris está SunOS, basado en la Versión 4 del Sistema V. Aparte del sistema operativo base, Solaris también incluye la interfaz gráfica de usuario llamada Open Windows, y el Common Desktop Environment (CDE), una colección de operaciones básicas denominada DeskSet y un entorno de trabajo en red llamado Open Network Computing. Solaris corre sobre estaciones de trabajo SPARC de Sun Microsystems y sobre computadoras personales 80x86 Intel.

Existe un gran número base de computadoras corriendo bajo Solaris. Sun ayudó a sacar a UNIX del reino de las microcomputadoras al entorno de escritorio, vendiendo sus primeros equipos en 1981. El SunOS original estaba basado en BSD.

En 1989 Sun empezó a trabajar con AT&T para unificar el SunOS, la Versión 3 del Sistema V UNIX y XENIX en la Versión 4; con Solaris, basado en SVR4, los usuarios podrán ejecutar el mismo sistema operativo tanto en sus estaciones de trabajo Sun como en sus PC.



Fig. 2.7 Logotipo del sistema operativo Solaris.

2.10 Interfaz de usuario

“*Open Windows* y el Common Desktop Environment (CDE) son la interfaz gráfica de usuario para Solaris. Están construidas siguiendo la especificación Open Look. Open Look es una especificación de interfaz de usuario, en vez de una incorporación, y fue originalmente desarrollado por Sun Microsystems apoyados por AT&T.”¹⁴

El Desk Set, un juego de operaciones empaquetado con Open Windows y CDE, ejecuta muchas tareas importantes, incluyendo algunas que son las tareas básicas de usuario. Por ejemplo, la Command Tool suministra una interfaz de línea de órdenes a SunOS. Se puede copiar, eliminar, mover y renombrar archivos utilizando el Administrador de Archivos. El editor de iconos permite crear iconos que se pueden ligar a archivos. La shell tool permite acceder a la línea de órdenes de SunOS para introducir órdenes.

El Desk Set incluye una variedad de útiles aplicaciones además de las básicas anteriormente descritas. Por ejemplo, el Administrador de calendarios permite mantener esquemas diarios, semanales mensuales y anuales y hacer listas para un grupo de usuarios en la red. La calculadora puede ser utilizada para realizar cálculos financieros y científicos.

¹⁴ Ídem. p. 1157

La herramienta Mail es una aplicación en el DeskSet que puede ser utilizada para crear y leer correo multimedia. Utilizando la herramienta Mail se pueden ligar datos de audio y vídeo a un e-mail. La herramienta de audio de Solaris suministra una GUI que permite grabar y reproducir archivos de audio, permitiendo mezclar diferentes archivos de audio y luego guardar el resultado. Esto permite a los usuarios ser productivos con un conjunto de aplicaciones integradas.

Solaris suministra la herramienta de administración para simplificar la mayoría de las tareas administrativas del sistema, la cual contiene las siguientes aplicaciones: el administrador de bases de datos, el administrador de impresión, el administrador Host y el administrador de cuentas de usuario. El administrador de bases de datos puede ser utilizado para manejar las tablas de servicio de información NetWork y archivos ufs en el directorio /etc; esto permite trabajar con una variedad de bases de datos de red. El administrador de impresión permite añadir una estación de trabajo a nuestra red o bien eliminar una estación de trabajo de ella.

El administrador de cuentas de usuario hace que resulte fácil añadir o eliminar usuarios de nuestro sistema, permite establecer contraseñas, crear el directorio propio de un nuevo usuario y establece permisos por omisión sobre los archivos de un nuevo usuario.

El DeskSet también suministra el Performance Meter (medidor de rendimiento), que puede ser utilizado para monitorear diez aspectos de nuestro sistema. Los elementos que pueden ser monitoreados aquí son: el porcentaje de CPU que se está utilizando, la actividad de paginación en páginas por segundo, el número de trabajos realizados por segundo, el número de interrupciones de trabajo por segundo, el tráfico de disco en transferencias por segundo, el número de conexiones de contexto por segundo, el número medio de procesos realizados en el último minuto, el número de colisiones por segundo detectadas en la Ethernet, el número de errores por segundo al recibir paquetes Ethernet por segundo.

2.11 Novedades de la versión de Solaris 2.7

A continuación resaltaremos algunas de las ventajas del sistema operativo Solaris.

El sistema operativo Solaris es el fundamento de la computación basada en Web. Es escalable y tiene la posibilidad de ejecutarse y desarrollarse en Internet. Análogo al tono de marcar telefónico, Solaris es el "tono de Web" de Internet.

El sistema operativo Solaris de 64 bits ofrece la capacidad, rendimiento y precisión necesarias para manejar archivos muy grandes, pues fue desarrollado para proporcionar capacidad, seguridad, interoperactividad, gestionabilidad, y conectividad global, mediante su servicio eficaz, consistente y pluri-facético.

2.12 Características del nuevo sistema operativo Solaris 7

Sistema operativo de 64 bits

El sistema operativo Solaris de 64 bits es un entorno de aplicación y desarrollo de 32 y 64 bits soportado por un sistema operativo de 64 bits. Esto permite la máxima compatibilidad e interoperatividad para las aplicaciones existentes, tanto las fuente como las binarias. Al mismo tiempo, el sistema operativo Solaris de 64 bits supera muchas de las limitaciones del sistema de 32 bits, en particular, el soporte de un espacio de direcciones virtual de 64 bits así como la eliminación de otras limitaciones del sistema de 32 bits (sólo para la edición SPARC).

Navegador de Web

El software de Solaris 7 se suministra con Netscape Communicator.

Administración de red y del sistema

- El registro de UFS es el proceso de almacenar las transacciones (los cambios que componen una operación UFS completa) en un registro antes de que se apliquen las transacciones al sistema de archivos UFS. Una vez que se ha almacenado una transacción, puede aplicarse ésta al sistema de archivos posteriormente.
- El registro de UFS proporciona dos ventajas: evita que los sistemas de archivos pierdan su coherencia, eliminando así la necesidad de ejecutar en algún momento el comando *fsck*; además, debido a que puede sobrepasarse *fsck*, el registro de UFS reduce el tiempo necesario para reorganizar un sistema si se produce una caída o después de un paro no completo.
- Otra ventaja es la opción de montaje de UFS `-o noatime`, para ignorar las actualizaciones de tiempo de acceso en los archivos; puede especificar la opción `-o noatime` al montar un sistema de archivos UFS. Esta opción reduce la actividad de disco en los sistemas de archivos en los que los tiempos de acceso no son importantes (por ejemplo, una cola de noticias).
- El Lightweight Directory Access Protocol (LDAP) es un protocolo de acceso de estándar abierto e independiente de la plataforma. Se ha diseñado para funcionar sobre TCP/IP y utiliza codificaciones cliente-servidor; la biblioteca de cliente incluida en esta versión permite a los desarrolladores escribir aplicaciones LDAP y a los usuarios ejecutar aplicaciones que permitan LDAP. El protocolo LDAP mejora la gestión de bases de datos de nombres.
- La reconfiguración dinámica permite al proveedor de servicios agregar, eliminar y/o sustituir, tarjetas de sistema de conexión en caliente en un sistema en ejecución, con lo que elimina el tiempo perdido durante el reorganizar (sólo en determinados sistemas SPARC).

- El comando *pgrep* consulta los procesos activos del sistema y muestra los identificadores de proceso de los procesos cuyos atributos concuerdan con los criterios especificados en la línea de comandos. El comando *pkill* funciona igual que el comando *pgrep* con la excepción que se envía la señal *kill* a cada identificador de proceso en lugar de mostrarlo por pantalla.
- La versión *sendmail* 8.9 está preparada para permitir la restricción de spam (correo electrónico masivo y no solicitado); cuenta con almacenamiento virtual, que permite la recepción del correo electrónico mediante nombres de dominio diferentes, y una jerarquía de configuración mejorada que hace que la creación de su propio archivo de configuración de *sendmail* sea mucho más sencilla.
- El software de Solaris 7 integra la conocida utilidad *traceroute*, que se utiliza para rastrear la ruta que sigue un paquete de IP hasta un sistema de Internet. Es especialmente útil para determinar las configuraciones erróneas del encaminamiento y los fallos de rutas de encaminamiento.
- La utilidad del volcado por caída del sistema incluyen las siguientes características:
 - El comando *dumpadm* permite a los administradores del sistema configurar los volcados por caída del sistema del sistema operativo.
 - Los datos de volcado se almacenan ahora en formato comprimido en el dispositivo de volcado.
 - Cuando un dispositivo de volcado forma parte de la configuración de volcado, la grabación de los archivos del núcleo central se hace en segundo plano.

Rendimiento de red

TCP con SACK. El reconocimiento selectivo de TCP (TCP SACK) proporciona el soporte para solucionar los problemas relacionados con la congestión y la pérdida de múltiples paquetes, especialmente en las aplicaciones que utilizan ventanas de TCP grandes a través de enlaces vía satélite o enlaces transcontinentales.

Seguridad de red

Se ha modificado RCP en base a GSS-API. Esto aumenta la integridad y confidencialidad de la seguridad y los servicios NFS dejan de estar unidos a un mecanismo de seguridad único o específico.

NIS+ mejora la seguridad de NIS+ al aumentar la longitud de la clave de autenticación de 192 bits a 640 bits.

Berkeley Internet Name Daemon (BIND); la implementación de DNS más conocida, se ha modernizado a la versión 8.1.2. Proporciona un nuevo archivo de configuración que mejora la seguridad de red mediante el uso de listas de control de acceso (LCA).

Instalación

Los programas de instalación de Solaris tienen una nueva casilla de verificación para la selección de la admisión de 64 bits; cuando se realiza la instalación en plataformas UltraSPARC está seleccionada de forma predeterminada.

El producto Web Star (en el CD de documentación) proporciona una interfaz de señalar y hacer clic con selecciones para instalar el servidor AnswerBook2, todas las colecciones de documentos del CD o una selección de colecciones de documentos.

En la versión 7 de Solaris, se han combinado en un solo CD las versiones adaptadas al entorno nacional inglesas y europeas. Como resultado de esto, hay disponibles más selecciones de entorno nacional durante la instalación de este CD.

Documentación

Las páginas del comando *man* están disponibles en formato AnswerBook2 en lugar de en formato AnswerBook. Esto proporciona mejoras en la navegación y enlace a las páginas del comando *man* directamente desde otros documentos de AnswerBook2.

Con un CD de documentación y acceso de usuario root al sistema al que está conectado el CD, puede ejecutarse directamente el servidor AnswerBook2.

El servidor AnswerBook2 puede ejecutarse sobre un servidor Web existente, tal como Sun WebServer, en lugar de necesitar ejecutar un servidor web adicional en el sistema únicamente para admitir AnswerBook2.

Soporte de idioma

El software de Solaris ha ampliado su soporte de Unicode con la adición de seis nuevos entornos nacionales UTF-8: francés, alemán, italiano, español, sueco y Europa. Además, se incluyeron los entornos nacionales Unicode ampliados con capacidad de escrituras múltiples. Los usuarios pueden introducir y mostrar texto con diferentes escrituras, como japonés, tailandés o ruso, y cambiar fácilmente entre ellas sin tener que cambiar o instalar un nuevo entorno nacional.

Se ha integrado el soporte de texto complejo para idiomas, como el árabe, hebreo y tailandés, que necesitan un preprocesamiento del texto especial para manejar el texto bidireccional, compuesto y contextual.

El software de Solaris 7 implementa Internet Intranet Input Method Protocol (IIIMP) para permitir la interoperatividad integrada entre los métodos de entrada proporcionados por las aplicaciones Solaris, Java y las no X Windows.

El programa de descarga de fuentes del escritorio permite a los usuarios descargar, eliminar, volver a codificar y convertir fuentes, comprobar el estado y realizar otras tareas administrativas en una impresora PostScript.

La Comunidad Económica Europea (CEE) acordó la estandarización de una única moneda, el Euro, a partir de enero de 1999; todas las empresas de cambio de moneda, banca y finanzas de la CEE empezaron a utilizar el euro en lugar de sus monedas locales. Para anticipar este cambio, el software Solaris 7 agregó el soporte para la moneda Euro con seis nuevos entornos nacionales de usuario.

El software de Solaris ha agregado el soporte para las regiones de Europa del Este, Tailandia y el Oriente Medio.

Estándares

El software Solaris 7 tiene la marca UNIX 98.

Gráficos/Imágenes

La biblioteca de imágenes de fundación XIL es adecuada para bibliotecas o aplicaciones que requieren imágenes o vídeo digital, como imágenes de documento, preparación previa de prensa en color, o generación y reproducción de vídeo digital.

La nueva admisión para la visualización de imágenes estereoscópicas permite la presentación de pares de imágenes que representan una vista de ojo izquierdo/ojo derecho. Esto proporciona una visualización de imágenes con percepción de profundidad.

El Kit de desarrollador de XIL se ofrece ahora de forma independiente a Solaris y se proporciona de forma gratuita.

Escritorio

Solaris Common Desktop Environment (CDE) contiene herramientas nuevas para buscar, manipular y gestionar las tarjetas de direcciones, aplicaciones, direcciones de correo electrónico, archivos, carpetas, sistemas, procesos y direcciones de Web de forma sencilla.

En CDE se incluye el soporte de Motif 2.1, que incluye cinco nuevos widgets de Motif y es seguro para multiproceso. Motif 2.1 admite los entornos nacionales de idiomas de texto complejo, estándares de ISO en los que un único binario desarrollado en el sistema operativo Solaris 7 proporciona el soporte estándar y avanzado para los usuarios de los idiomas hebreo, árabe y tailandés.

Impresión

El programa de descarga de fuentes permite a los usuarios descargar, eliminar, volver a codificar y convertir fuentes, comprobar el estado y realizar otras tareas administrativas en una impresora PostScript.

Entorno de desarrollo de software

El sistema operativo Solaris 7 proporciona a los desarrolladores entornos completos de desarrollo de 32 y 64 bits.

El enlazador en tiempo de ejecución permite a los programas encontrar bibliotecas compartidas sin tener que definir *LD_LIBRARY_PATH* y hace que su carga sea todavía más eficaz.

El Kit de desarrollo de Java 1.1.5 para Solaris se ha ajustado y probado especialmente. Como resultado, ofrece una escalabilidad y rendimiento mejorados significativamente para las aplicaciones Java desarrolladas e integradas en la empresa y a través de la red.

RESUMEN CAPÍTULO II

El Sistema Operativo UNIX es uno de los sistemas operativos más ventajosos y utilizados en la actualidad; potente, flexible y versátil, este sistema puede ser utilizado en todo tipo de computadoras, lo que le augura mayor y continuo crecimiento, pues ofrece grandes posibilidades, como adaptabilidad y simplicidad, así como amplia posibilidad de adecuación a redes. Además, este sistema operativo es multiusuario, es decir, puede ser utilizado por uno o varios usuarios, y multitarea, es decir, que un único usuario puede realizar más de una tarea al mismo tiempo.

Este sistema operativo presenta ciertas características que le han permitido su desarrollo. Una de ellas es el código fuente abierto, que, utilizando un código sencillo, modular y compacto, y estando disponible a usuarios y programadores, ha permitido la adaptación del sistema a múltiples formas.

El sistema UNIX ofrece al usuario diferentes herramientas y utilidades para realizar gran variedad de trabajos diseñados para operar en conjunto, como partes de una máquina o bloques de construcción. Igualmente, este sistema es de gran utilidad en el entorno de redes por sus posibilidades como cliente/servidor, pues tiene la facultad de procesar información compartida en red por diferentes computadoras, además de ofrecer aplicaciones combinadas. Además, este sistema ofrece gran posibilidad de ser portado a otras máquinas con sistemas operativos diferentes a él, pues presenta menos complicaciones para ser adaptado y ejecutado en máquinas nuevas.

Los programas de aplicación del sistema UNIX realizan diferentes tipos de tareas; existen dos tipos de aplicaciones, las horizontales y las verticales; las primeras pueden ser empleadas por una variedad de usuarios de diferentes sectores, pues incluyen programas versátiles, mientras que las segundas son específicas de usuarios de la industria. Existen otras aplicaciones de red, que permiten ampliar los servicios disponibles en Internet, como los exploradores (browsers), además de aquellas que permiten a los usuarios crear y visualizar archivos de multimedia: sonido, imagen y video.

UNIX tiene múltiples utilidades o programas de usuario, como las órdenes o herramientas que se pueden utilizar de forma independiente o conjunta para realizar las tareas asignadas. Este sistema suministra utilidades estándar, con múltiples funciones, aplicaciones y propósitos, además de ofrecer una vasta cantidad de utilidades opcionales que se pueden adquirir por diferentes medios.

En el tema 2.3 se presentó una síntesis de la historia del surgimiento y evolución del sistema operativo UNIX, su aparición en los años sesenta y cómo fue evolucionando a través de las décadas, perfeccionándose y prosperando con diferentes denominaciones (MULTICS, UNICS, y, en 1970, UNIX). En la década de los setenta, este sistema se desarrolló en diferentes versiones, para ser comercializado en los ochenta por AT&T, que, en 1983, homogenizó su uso y dió pie a la compatibilidad con futuras versiones del mismo sistema, así como la aparición y perfeccionamiento de nuevas características de disposición de recursos físicos y de aplicaciones, lo que permitió que en los años posteriores se internacionalizara y cobrara la importancia y trascendencia que tiene en nuestros días.

En este proceso surgieron otros sistemas complementarios a UNIX, con diseño, funciones y estructuras diferentes, pero siguiendo el mismo principio y reportando grandes ventajas para los programadores, operadores, las empresas y la industria cibernética en su conjunto.

Por otro lado, con el objetivo de definir un estándar de sistema operativo de UNIX, que asegurara a los usuarios de este sistema la portabilidad de las aplicaciones a otros sistemas, se estableció en 1984 un estándar de trabajo, que al año siguiente se fusionó a los estándares del proyecto P1003 y se incorporó a los estándares POSIX, que es un grupo de estándares sobre las aplicaciones del sistema operativo, que abarca una gran cantidad de las herramientas indispensables en las diferentes versiones del sistema, así, por ejemplo, existen nombres específicos para cada estándar de aplicación: P1003.1, P1003.2, 1003.3, etcétera.

UNIX Sistema V Versión 4 surgió de la fusión del conjunto de instrucciones del sistema V versión 3.2 y las instrucciones más ordinarias de los sistemas BSD, XENIX y SunOS, para agregar y actualizar órdenes. Esta versión soporta varios shells: por omisión, estándar, job, korn y C, que ofrecen diferentes funciones y ventajas al ser ejecutados.

Además, esta versión reestructuró la disposición del árbol de directorio del sistema UNIX, agregándole a la disposición de los archivos en directorios jerárquicos, aplicaciones para adecuar los entornos de red y compartir archivos remotos. Asimismo, la Versión 4 añade aplicaciones que facilitan el empleo de dispositivos nuevos, como el Device-Kernel (DKI), e implanta una interfaz para ingreso al sistema.

A diferencia de versiones anteriores de UNIX, la versión 4 presenta nuevas características para soportar el procesamiento de aplicaciones en tiempo real, es decir, la posibilidad de ejecución de programas de la computadora en intervalos puntuales de tiempo. La versión 4 del sistema V UNIX comprende procesos tanto de tiempo compartido como de tiempo real, encargándose de estos últimos mediante las llamadas al sistema, que incluyen ciertas adaptaciones al sistema BSD.

Esta versión también incluye otras ventajas que facilitan la administración y mantenimiento del sistema, como la simplificación de la interfaz de menú, las mejoras en los procedimientos de seguridad y restauración y la simplificación de la instalación del software.

Dada la importancia de las redes para esta versión, se le incorporaron nuevas capacidades de red de los sistemas BSD y SunOS que contienen el paquete Internet TCP/IP, el cual comprende órdenes de transferencia de archivos y presentación y ejecución remota. Estas modificaciones han permitido incorporar características de red, como el sistema de archivos de red (NFS), el protocolo estándar de industria (RPC) y el de representación externa datos (XDR).

Aparte de lo mencionado, esta versión se diferencia de las anteriores al ofrecer interfaces de usuario estándares y sistema de ventanas para aplicaciones basadas en carácter y gráficos; existen dos interfaces gráficas principales: OPEN LOOK y MOTIF, que contienen algunas diferencias en aplicaciones, usos y popularidad, además de la interfaz gráfica estándar CDE; por otro lado FACE es una interfaz de carácter para terminales de usuario.

La versión 4 del sistema V de UNIX es uno de los sistemas más empleados, pues la mayoría de las otras versiones de UNIX se basan en él, al ajustarse a sus estándares y compartir muchas de sus características. Es así que existen múltiples variantes de este sistema que denominan fusiones, productores de sistema, aplicaciones específicas y otras particularidades, como UNIXWARE (Novell NetWare), UNIX SCO (Santa Cruz Operation), IRIS (Silicon Graphics), HP-UX (Hewlett-Packard), AIX (IBM), LINUX (Linus Tovalds), y A/UX (Apple's).

La estructura del sistema operativo UNIX está conformada por varios elementos, entre ellos el núcleo, el shell, el sistema de archivos y los órdenes (programas de usuario). Este sistema está perfilado para operar la información comprendida en discos, por ello organiza la información mediante la localización de los archivos a través de una estructura jerárquica de directorios denominada estructura árbol, que controla todos los archivos de sistema agrupándolos por características o funciones afines; esta estructura permite que los archivos no se obstruyan entre sí.

Un archivo es la estructura básica que emplea UNIX para almacenar información, asignándole a cada uno cierto número de identificación que apunta a la tabla de i-nodos; este sistema organiza los archivos en registro o bloques de longitud fija, siguiendo la tradicional organización de datos en el disco. Los programas de este sistema tienen libertad de organizar sus archivos según el modo en que sus datos estén recopilados en el disco.

Por su parte, un directorio mantiene un comportamiento similar a los archivos pero con registros de longitud fija; su contenido apunta a unas tablas de i-nodos separadas de la estructura, desde las cuales se apunta a los bloques físicos de los archivos en el disco.

Al conjunto de trabajo conformado por la estructura de archivos, directorios y tablas de i-nodos se le conoce como sistema de Archivos o File System, que reside en un soporte físico que es el disco magnético; si el disco es de gran capacidad, se divide en varios discos lógicos o particiones que funcionan como archivos independientes; cada uno de estos sistemas dispone de su propia tabla de i-nodos; el directorio más importante del File System es el directorio raíz.

Las características de un sistema de archivos son: estructura jerárquica, tratamiento consistente a los datos, facultad de crear y borrar archivos, protección de datos y tratamiento de archivos periféricos como archivos. El file system está compuesto por un bloque de carga (arranque), un súper bloque, lista de i-nodos y bloques de datos. Al tener una estructura jerárquica, el File System permite agrupar lógicamente la información de los usuarios y manejar como una unidad a un conjunto de archivos.

La manera en que los usuarios se "comunican" con la computadora es por medio de un shell, pues es el intermediario entre el sistema operativo básico y el usuario; hay diferentes tipos de shell, el estándar Bourne Shell, el C Shell, el Korn Shell, entre otros. El shell de UNIX puede crear variables del entorno según las características establecidas por el administrador del sistema; estas variables definen el entorno de trabajo y pueden ser redefinidas por el usuario.

Otra parte de la estructura de sistema operativo UNIX es el núcleo, que interactúa directamente con el hardware de la computadora por medio de los controladores de dispositivo incorporados a ella; sus principales funciones son: gestión de la memoria, control de acceso al equipo, mantenimiento del File System, manejo de interrupciones, manejo de errores, realización de servicios de E/S y asignación de recursos de la computadora. Los programas interactúan con el núcleo por medio de las llamadas al sistema.

El surgimiento del sistema operativo SunOS (1991), estuvo a cargo de Sun Microsystems y se basa en UNIX; este sistema se fue modificando y perfeccionando hasta llegar a su versión actual: Solaris 2.7, que representa grandes ventajas para los usuarios del sistema UNIX, como la interfaz gráfica Open Windows, el CDE, algunas operaciones básicas de DeskSet y un entorno de trabajo en red denominado Open Network Computing. En la actualidad diferentes empresas cibernéticas se han asociado en el desarrollo de el sistema Solaris, que es uno de los más utilizados en nuestros días.

Solaris ofrece el Open Windows y el CDE como interfaces de usuario que cumplen con la especificación de interfaz de usuario Open Look, así como el juego de operaciones DeskSet, que ejecuta múltiples tareas básicas de usuario, e incluye una variedad de aplicaciones útiles al usuario.

Una de las principales ventajas del sistema operativo Solaris es que es el sostén de la computación basada en Web, además, es escalable y puede ejecutarse desde Internet. Este sistema es idóneo para el manejo de archivos de gran tamaño, proporcionando capacidad, seguridad, interoperatividad y conectividad global.

Algunas de las características del sistema operativo Solaris 7 son: su número de bits (64), el software que lo suministra (Netscape Communicator), sus virtudes de la administración de red y del sistema (el proceso UFS, el protocolo LDAP, los comandos pgrep, pkill, dumpadm, entre otros), lo que le asegura el rendimiento de red por tener la capacidad de soporte para solucionar diferentes problemas, así como el sistema de seguridad de red que aumenta la integridad y confidencialidad de los servicios.

Una de las principales características de la instalación de Solaris es que en su versión 7 combina en un solo CD las versiones adaptadas de diferentes naciones, lo que aumenta su uso y disponibilidad. Asimismo, esta versión del sistema proporciona diferentes mejoras en los servicios, entre ellas se encuentran la documentación y acceso de usuario, el amplio soporte de Unicode, con la adición de nuevos idiomas, escritura múltiple y soporte de texto complejo; así como las innovaciones en sus aplicaciones de gráficos e imágenes en diferentes modalidades.

También cuenta con nuevas herramientas de escritorio, como manipulación y gestión de tarjetas de direcciones, aplicaciones, direcciones de e-mail, archivos, carpetas, sistemas, procesos y direcciones Web, mediante el CDE que incluye soporte Motif 2.1, y las tareas administrativas de impresión por medio del programa de descarga de fuentes para los usuarios con todas sus ventajas.

CAPÍTULO III

III. ADMINISTRACIÓN DEL SISTEMA OPERATIVO SOLARIS 2.7.**3.1 Responsabilidades y labores del administrador**

El sistema operativo (SO) utilizado en los equipos SUN es el llamado Solaris, que es la versión particular de UNIX implementada por SunSoft. Es un sistema multiusuario y multitarea que permite controlar y anexar diversos dispositivos al equipo, como impresoras, discos, cámaras de vídeo, unidades de cinta, etc. Además, permite conectar el equipo a una red y de esta forma, compartir sus recursos con otros.

Todas estas ventajas que brinda el SO, traen consigo una serie de responsabilidades para establecer, mantener y solucionar cualquier problema que se presente con su funcionamiento. A este conjunto de responsabilidades se les conoce como administración del sistema.

El administrador de sistema es el encargado de administrar los recursos del equipo que tiene a su cargo (Fig. 3.1), por lo que es responsable de la instalación y configuración del Sistema Operativo, mantenerlo en buenas condiciones y repararlo cuando presente fallas. Es decir, sus responsabilidades son todas aquellas tareas que se encuentran fuera del alcance de los usuarios comunes, ya sea porque se requiere de privilegios especiales dentro del sistema o porque tiene que ver directamente con el manejo del equipo (instalación o remoción de partes).

En forma más específica, el administrador tiene que realizar lo siguiente:

- Instalación y pruebas del hardware. Para ello se requieren conocimientos del funcionamiento y características de los diversos componentes físicos que integran al equipo, lo que permitirá armarlo e instalar periféricos sin mayores dificultades.
- Planeación. Determinar la distribución de los discos, los usuarios y la carga de trabajo que soportarán los equipos.
- Instalación del Sistema Operativo.
- Configuración. Realizar las adecuaciones necesarias para que funcione en óptimas condiciones.
- Operación. Operaciones rutinarias que van desde encender y apagar el equipo, solucionar cualquier problema que se presente, añadir y controlar usuarios, así como instruirlos para que puedan optimizar el uso de los recursos, implantar reglas de uso, instalar nuevo software, etc.
- Administración de recursos. Vigilar el uso de CPU, espacio en disco, frecuencia de utilización de aplicaciones, base de datos, etc, para poder llevarlo y controlarlo en forma adecuada.
- Instalación y configuración en red. Seleccionar, conectar y configurar el equipo al tipo de red más adecuado, ponerse en contacto con el administrador de redes para establecer parámetros del equipo y políticas de uso.
- Seguridad. Monitorear continuamente el uso del equipo y recursos para mantenerlo al máximo, implementar medidas y activar utilerías de seguridad, lo cual implica conocer profundamente el equipo y el Sistema Operativo.

- Análisis de fallas. Llevar una bitácora del funcionamiento y fallas del equipo; porque de ahí se pueden deducir y atender a tiempo problemas mayores antes de que ocurran.
- Respaldos. Respaldar la información que se encuentra almacenada en el equipo, ya que es lo más valioso, de ahí la importancia que tiene la labor de respaldar y, en su caso, restaurar la información almacenada.
- Actualizaciones. Realizar un análisis antes de instalar una nueva versión de cualquier software; en el cual se determinen las ventajas, compatibilidad con los archivos y sistema actual, los problemas que pueda tener, etc.
- Asistencia en dudas sobre el manejo del sistema. Debe tener los conocimientos suficientes para poder brindar la asesoría necesaria a los distintos usuarios del equipo, o por lo menos, poder guiarlos en la búsqueda de soluciones.

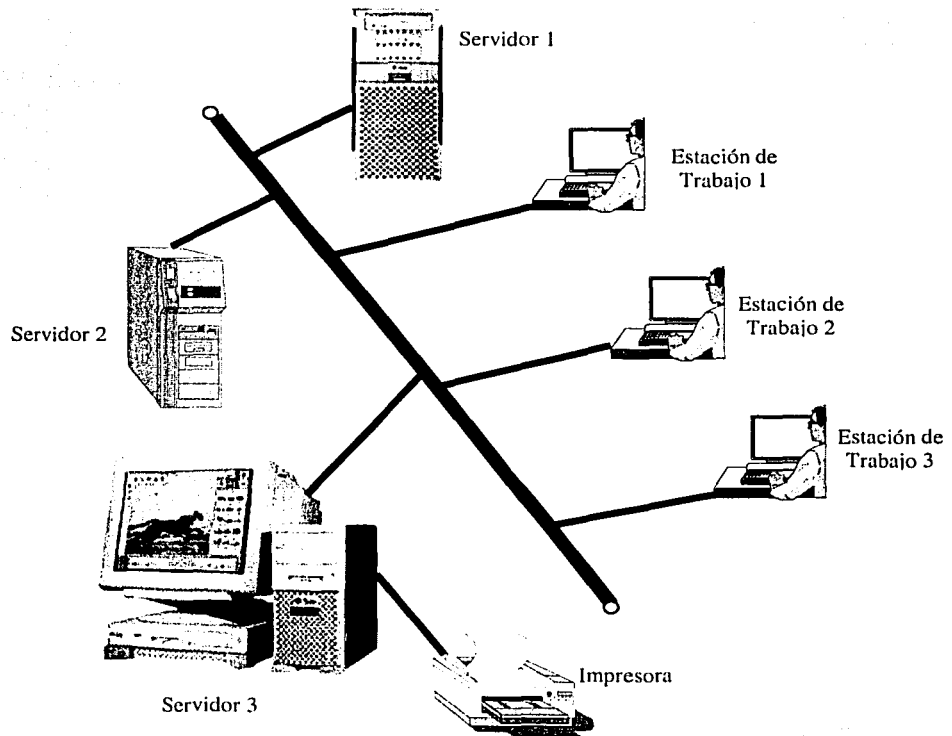


Fig. 3.1 Ejemplo de la labor del administrador del sistema.

3.2 Instalación

La instalación es el proceso durante el cual son colocados todos los componentes de un equipo en condiciones que permita su funcionamiento adecuado. Ésta la podemos dividir en dos partes:

- Instalación del hardware, en la cual se acondicionan y conectan todos los componentes físicos del equipo.
- La instalación del software, en la que son colocados dentro de la máquina todos los programas necesarios para poder operarla correctamente.

3.3 Instalación del hardware

La instalación del hardware es un proceso relativamente sencillo, pero se tiene que realizar con cuidado. De ello depende la vida útil del equipo así como su buen funcionamiento. Existen varios puntos en los cuales se debe poner atención.

- Selección del lugar.
- Adecuación de las instalaciones.
- Conexión de los componentes del sistema.
- Tener acceso restringido al lugar donde se encuentra el equipo para que solo sea usado por el personal autorizado.
- Para evitar un calentamiento excesivo del equipo se debe tomar en cuenta las siguientes indicaciones:
 - o Colocar sobre una superficie plana y firme.
 - o No colocar carpetas ni manteles debajo de él, ya que esto interfiere con la circulación de aire por la parte inferior del gabinete y genera un calentamiento mayor.
 - o No debe ser colocado en lugares cerrados, ni cubierto con fundas mientras esté encendido.
 - o Procurar dejar el espacio suficiente a su alrededor para que exista una adecuada ventilación; si es colocado junto a una pared, se recomienda dejar una separación como mínimo de 15cm. entre el equipo y la pared.
 - o Evitar colocarlo cerca de ventanas donde puedan alcanzarlo los rayos solares.

Una vez seleccionado el lugar, éste se debe acondicionar. Es necesario poner especial atención en tres factores:

- Sistema de alimentación de energía.
- Fuente de respaldo de energía (UPS: Uninterruptible Power Suply -Fuente de Poder Ininterrumpida-). Esto con el fin de que si se produce un fallo en la energía eléctrica el equipo no sufra un apagón inesperado que pueda dañar al software o al hardware.
- Un sistema de aire acondicionado que permita mantener al equipo a baja temperatura ya que si no se cuenta con ello el equipo podría sobrecalentarse y sufrir daños.

3.4 Conectar los componentes del sistema

Esta es la última fase en el proceso de instalación física, en ella se debe comenzar por inspeccionar el estado inicial del equipo, hasta terminar con la conexión de todos los componentes; de tal forma que se encuentre listo para ser encendido.

Adicional a los dispositivos básicos se pueden adquirir otros periféricos que permitan mejorar el funcionamiento del sistema, como son: unidades de disco, CD-ROM, cinta y cartucho, controladores de discos ópticos, impresoras, modems, etc.

Es importante recordar que siempre se deben leer todos y cada uno de los documentos que acompañan al equipo, en especial, antes de realizar la instalación: el de Material Packing List, que contiene la lista de todos los componentes que vienen en la caja, tanto manuales, equipo o documentos, se pueden detectar piezas faltantes; el de Readme Before Installing, que describe una serie de pasos y medidas que se deben tomar antes de realizar la instalación; el de Owner's Guide, que describe las características del equipo y los pasos para instalar diversos componentes opcionales; así como cualquier hoja de actualización, que puede contener información de último momento y que no se encuentra dentro de los manuales, o descripción de erratas de los documentos.

Un buen administrador debe estar acostumbrado a leer éstos y todos los documentos disponibles para el equipo, ya que una muy importante función que tiene es la de mantenerse actualizado respecto a las capacidades, características y funcionamiento, tanto del hardware como del software, que se encuentre disponible en él.

3.5 Conexión física

En la parte posterior del equipo Sun, están los distintos puertos con que cuenta y los periféricos que deben ser colocados en cada uno de ellos. Los puertos restantes son para dispositivos adicionales que pueden ser conectados al equipo cuando se adquieran, pero uno de ellos merece especial atención, el del puerto SCSI. A diferencia de los demás en los cuales sólo se puede conectar un dispositivo, éste es un bus al cual se pueden conectar hasta 7. Ver la fig. 3.2.

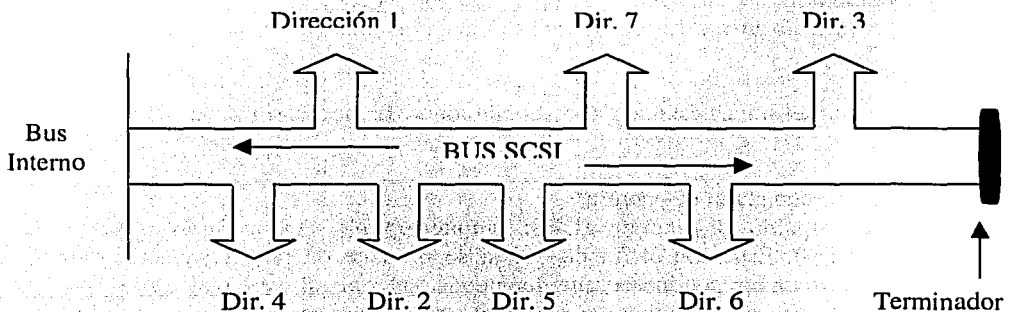


Fig. 3.2 Diagrama de la arquitectura del BUS SCSI.

La información viaja por el bus y todos los dispositivos que estén conectados a él la pueden percibir, por lo que para distinguir a quién va dirigida, se utiliza una dirección. Cada dispositivo SCSI cuenta con un método que le permita definirla: unos utilizan puentes, otros interruptores en miniatura (mini dip switches), o también botones selectores que permiten cambiar la selección actual.

Las direcciones que se utilizan son de la 0 a la 7, pero la 7 está reservada para direccionar a la tarjeta controladora SCSI en sí y, por acuerdo, la dirección 0 se da al disco interno donde se coloca el sistema operativo y que sirve de arranque, mientras la 1 se asigna al segundo dispositivo interno. Por tanto, se pueden conectar hasta 6 dispositivos externos con direcciones de 1 a 6, sin que tengan una dirección repetida.

Las unidades externas son conectadas una a continuación de la otra formando una cadena, al final se debe conectar un terminador, que tiene la función de impedir que la información siga viajando indefinidamente por el bus. La longitud máxima permitida es de 2m. Si no es conectado ningún dispositivo externo, se debe colocar el terminador en el puerto SCSI.

La importancia de realizar la conexión correctamente y poner al final el terminador, reside en lo siguiente: al encender el equipo, una de sus funciones es la de revisar todos los dispositivos que se encuentren conectados al bus, así como reconocer sus direcciones para poder establecer comunicación con cada uno de ellos, si alguno no está encendido al momento de realizar estas pruebas, el sistema no lo reconocerá después, dado que estas pruebas se realizan únicamente al encender el equipo. Por este motivo se deben encender en primer lugar, los dispositivos externos que se encuentren conectados al bus y al final el equipo.

Una vez que la unidad sea reconocida por el sistema, ésta se puede apagar y encender cuando sea necesario y el sistema la seguirá reconociendo. La única forma de que el sistema reconozca a un equipo que no estaba encendido es la de dar de baja el equipo, apagarlo y volverlo a encender, pero en esta ocasión el dispositivo debe estar encendido.

Al igual que la computadora, cada uno de los dispositivos externos al momento de ser encendidos, realizan una serie de pruebas en forma independiente para determinar su estado y quedar finalmente en operación. Dependiendo del dispositivo, puede que la prueba sea casi instantánea o que dure algunos minutos, y esta información se puede ver en el manual de cada uno de ellos. Por este motivo se debe dar el tiempo suficiente para que todos los periféricos estén listos antes de encender la computadora. Si se desconoce el tiempo de prueba que requiere alguno, es recomendable encenderlo y esperar a que la luz del led sea estable, dar un lapso de más y encender el equipo.

Finalmente, una vez conectados todos los dispositivos al CPU, éstos deben ser conectados a una fuente de alimentación de energía regulada. Efectuar una última revisión a las conexiones y encenderlos. La regla general de secuencia de encendido es:

- Si existen dispositivos externos, encenderlos.
- Dar un tiempo para que arranquen y queden en operación.
- Encender el monitor.
- Encender el CPU.

3.6 OpenBoot PROM

Todos los sistemas Sun tienen programas residentes que proveen de pruebas básicas al hardware, prioridades de inicialización para el arranque y además permiten inicializar el sistema desde un amplio rango de dispositivos. Es un programa de solo lectura abierto ya que dispone de opciones que permiten manipular el hardware mediante el cambio de parámetros siendo a la vez una interfaz de usuario.

Elementos básicos

Los elementos básicos del hardware Boot PROM en los sistemas Sun son:

-El Boot PROM es encontrado dentro de la tarjeta principal del sistema. (Fig. 3.3)
-La non-volatile random access memory (NVRAM) es un chip que contiene información importante sobre el sistema como:

EEPROM: Se encuentran los parámetros guardados de la NVRAM.

Time of day: También contiene la fecha y hora del sistema.

Ethernet address: Éste es el número de registro único del equipo en hexadecimal que le permite al protocolo TCP/IP darle un solo registro de identificación a nivel mundial.

Hostid: Es el número de Ethernet transformado a decimal para identificar el equipo.

Battery: Esta es una batería dentro de la NVRAM que provee de energía para que los datos no sean borrados, aún cuando este apagado el equipo.

Contenido del Boot PROM

- Power-on self-test (POST): Corre pruebas de diagnóstico al encender el equipo.
- Device drivers: Este es el manejador de dispositivos el cual se encarga de controlarlos y manipularlos.
- User interfaces: Provee de una interfaz de usuario para manipulación.
- Default Parameters: Contiene la NVRAM que es donde se almacenan los cambios de parámetros.

Non-volatile RAM (NVRAM)

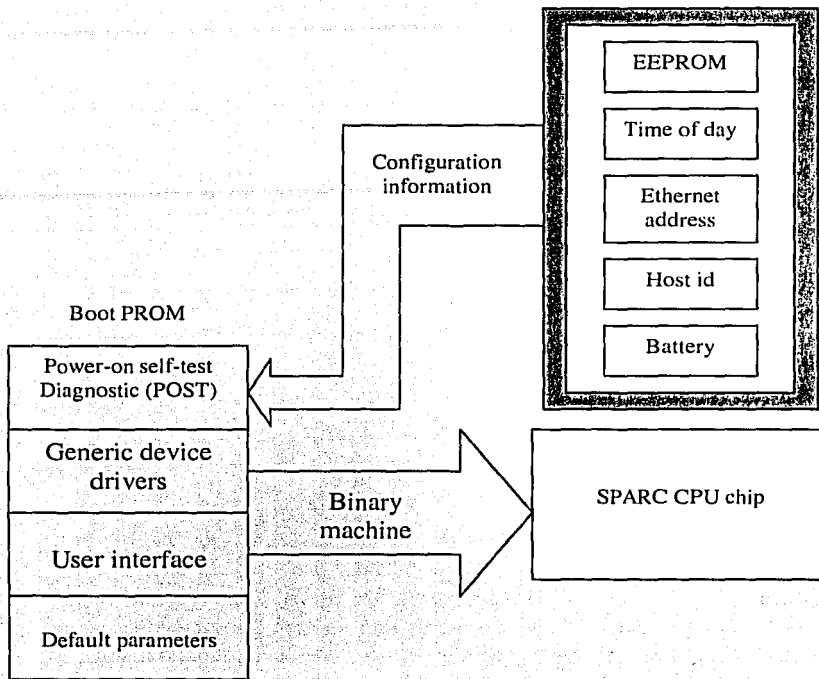


FIG. 3.3 Relación de elementos básicos entre la NVRAM y el CPU.

Comandos básicos del Boot PROM

La interfaz de usuario del Open Boot es llamada el PROM Monitor. El prompt que se ve en el nivel del monitor es un ok, este prompt se ve solo cuando el sistema operativo Solaris no está corriendo. El prompt reconoce la totalidad de los comandos del Open Boot, los siguientes comandos son los más comúnmente usados por el Open Boot:

- ok banner
- ok boot
- ok help
- ok printenv
- ok setenv
- ok set-defaults
- ok probe-scsi

- **ok probe-scsi-all**
- **ok probe-ide**
- **ok reset**

El comando banner

El comando banner lista la información mas importante del sistema como el modelo, cantidad de memoria, y el hostid. También se puede usar el comando banner para identificar el número de la versión del PROM. Ejemplo del comando banner:

ok banner

```
Sun UltraSPARC 60 UPA/PCI (2 X UltraSPARC-II 296MHz), Keyboard Present
Open Boot 3.11, 256 MB memory installed, Serial #3159808.
Ethernet address 8:0:22:1:e7:3f, Host ID: 72303700.
```

Ok

El comando boot

El comando boot sirve para inicializar el sistema operativo. Una vez que se enciende el equipo, el sistema nos deja en el PROM monitor, para cargar el sistema operativo se le introduce el comando boot. Se puede colocar una o más opciones al final de la cadena del comando afectando en el proceso de la carga del sistema operativo.

Las opciones más usuales del comando boot son las siguientes:

- a – Interactivo: Permite interactuar con el inicio del sistema operativo.
- r – Reconfigura: Cuando se conectan nuevos dispositivos en el equipo con el comando boot y la opción -r, el sistema crea la nueva estructura de los dispositivos conectados al equipo.
- s – Modo simple usuario: Este modo no carga los servicios de red, esto se puede utilizar cuando existen problemas con el equipo.
- v – Verbose: Despliega mensajes detallados cuando se esta haciendo la carga del sistema operativo mostrando cada uno de los pasos que se van realizando durante la misma, puede utilizarse para detectar problemas con el sistema.

Sintaxis: **ok boot** -opciones

El comando help

El comando help permite obtener las categorías de ayuda acerca de los diferentes comandos del PROM; para obtener ayuda más detallada se puede introducir help y el nombre del comando, o help y la primera palabra del nombre de la categoría. Ejemplo del comando help:

ok help

Main categories are:

Repeated loops

Defining new commands

Numeric output

Radix (number base conversion)

Arithmetic

Memory access

Line editor

System and boot configuration parameters

Select I/O devices

Floppy eject

Power on reset

Diag (diagnostic routines)

Resume execution

File download and boot

nvrarc (making new commands permanent).

Cambio del dispositivo de inicialización por default

En el Open Boot PROM se encuentran variables que pueden ser modificadas para cambiar el dispositivo de arranque por default.

El comando printenv, despliega las variables que se encuentran cargadas en la NVRAM permitiendo conocer los valores por default de las variables con que inicializa el sistema. Ejemplo de algunos valores:

ok printenv

Parameter Name Value Default Value

tpe-link-test? True true

output-device ttya screen

input-device ttya keyboard

sbus-probe-list f0123 f0123

keyboard-click? False false

diag-file

boot-file

boot-device disk disk

auto-boot? true true

More [<space>,<cr>, q] ?

El comando `setenv` redefine las variables de medio ambiente del PROM. Ejemplo:

```
ok printenv auto-boot?
auto-boot? true true
ok
ok setenv auto-boot? false
auto-boot? = false
```

El comando `set-defaults` permite regresar a los valores definidos por default en el sistema:

```
ok set-defaults
```

Setting NVRAM parameters to default values.

El bus SCSI

Asignación de direcciones en la tarjeta

La tabla de la figura 3.4 es una lista convencional de asignación de direcciones SCSI. Para manejadores del primer SCSI host adaptador. SCSI I y II soportan 8 dispositivos que son más comúnmente encontrados sobre sistemas 1.x y 2.x de las versiones del PROM.

La diferencia entre SCSI I y II y el III, es que el tres puede tener hasta 15 dispositivos conectados a él dentro de la misma controladora y este sólo se encuentra en versiones del PROM 3.x en adelante, no hay otra versión.

Tarjeta	0	1	2	3	4	5	6	7	8-15
Versión del PROM									
1.x	disco	disco	disco	Boot disco	tape	tape	cdrom	SHA	Otro
2.x	disco	disco	disco	Boot disco	tape	tape	cdrom	SHA	Otro
3.x	Boot disco	disco	disco	disco	tape	tape	cdrom	SHA	Otro

Fig. 3.4 Lista convencional de asignación de direcciones SCSI.

Tarjeta de direcciones en el ambiente Sun

El direccionamiento asocia a un manejador de dispositivo con un dispositivo periférico. Este permite al kernel identificar al dispositivo. Solaris 7 provee una asignación dinámica de controladores, esto permite al administrador del sistema colocarle un número de dirección única, instalar el periférico y tener la configuración del kernel con el manejador de dispositivo correcto.

El número de host adaptador es asignado automáticamente por el sistema operativo basado sobre la tarjeta o la localidad. La figura 3.5 muestra el ejemplo de un host SCSI adaptador, controlador de disco y particiones del disco.

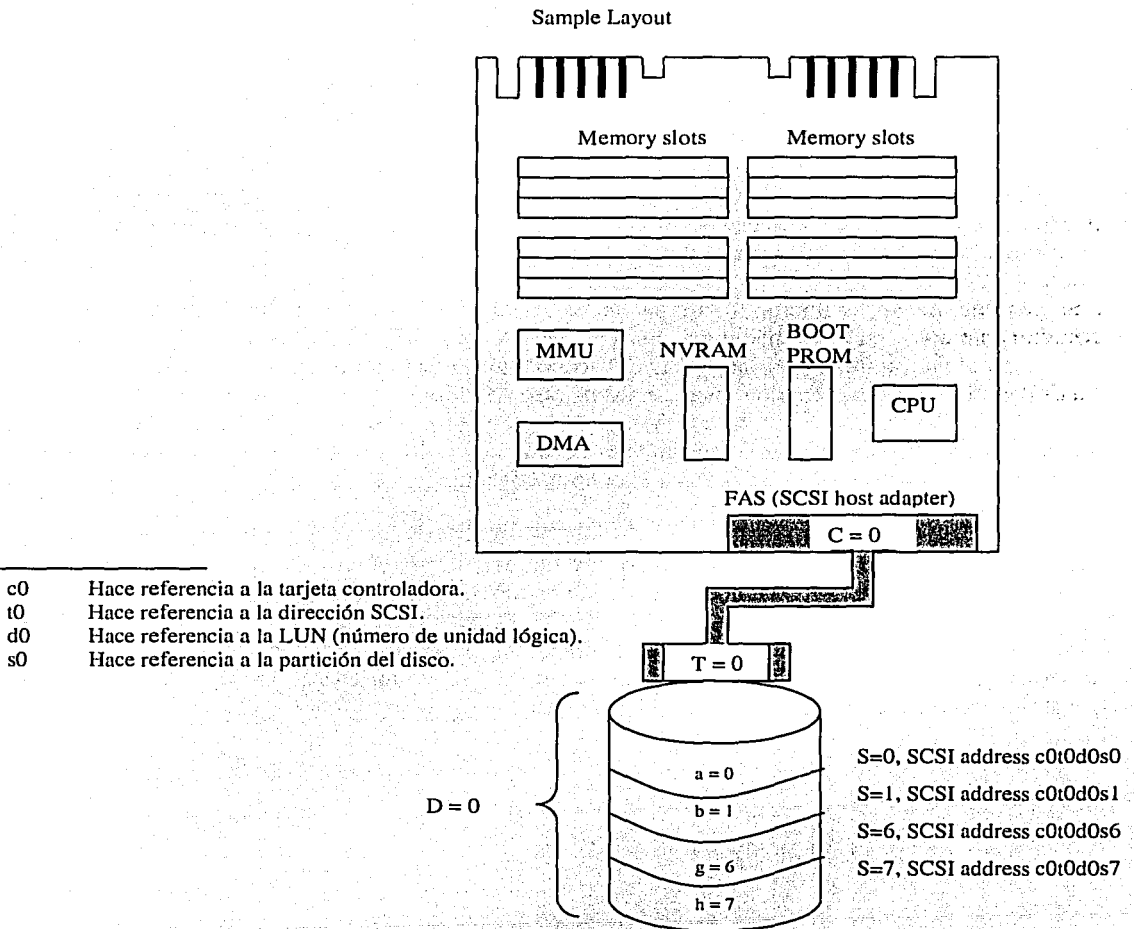


Fig. 3.5 Ejemplo de un adaptador SCSI.

3.7 Instalación del sistema operativo

Antes de instalar el sistema operativo se necesita saber qué tipo de configuración se desea para el equipo. Las configuraciones existentes son:

- Clientes del sistema: Esta configuración puede ser de tres maneras, Disklees, Autoclient y JavaStation, para las cuales se necesitan recursos de otra máquina para iniciar y operar, en este caso de un servidor que le provee los recursos.
- Disklees: Un cliente de estos es una estación de trabajo sin disco, necesita de otro sistema para cargar los programas necesarios de arranque, para utilizar espacio en disco y correr procesos, consta solo del teclado y monitor.
- Autoclient: El Autoclient es similar al cliente Disklees solo que ésta configuración sí tiene un disco que utiliza como área de swap en donde temporalmente puede guardar los file systems de root y usr pudiendo utilizarlos de ser necesario.
- JavaStation: Este tipo de cliente ha sido desarrollado para "cero administración"; toda la administración es hecha por el servidor por medio de la red local incluyendo la inicialización, configuración y almacenamiento de datos.
- Standalone: La configuración Standalone no necesita soporte de otras máquinas para iniciar; puede acceder a archivos en otras máquinas si se encuentra en una red, tiene su propio disco y puede tener conectada alguna unidad de cinta o de CD-ROM junto con sus controladores. Esta configuración hace al equipo independiente de los recursos de otro servidor.
- Server: La configuración de un servidor es normalmente un sistema que posee software adicional para soportar trabajo en red, se pueden encontrar conectados a él diferentes máquinas que obtienen recursos del mismo.

Un servidor puede tener discos locales o arreglos de discos externos y por lo general tiene una unidad de cinta y de CD-ROM, además de algunos otros componentes posibles.

Como se puede ver cada una de las configuraciones tiene características propias que pueden ser utilizadas de acuerdo a los requerimientos específicos del administrador. En este capítulo de tesis se mostrará cómo instalar el modo de servidor o server.

Grupo de software

Para la instalación del S. O. También es necesario conocer qué son los paquetes, los clusters y la configuración clusters ya que la instalación del software es compuesta de estas tres partes, (Fig. 3.6)

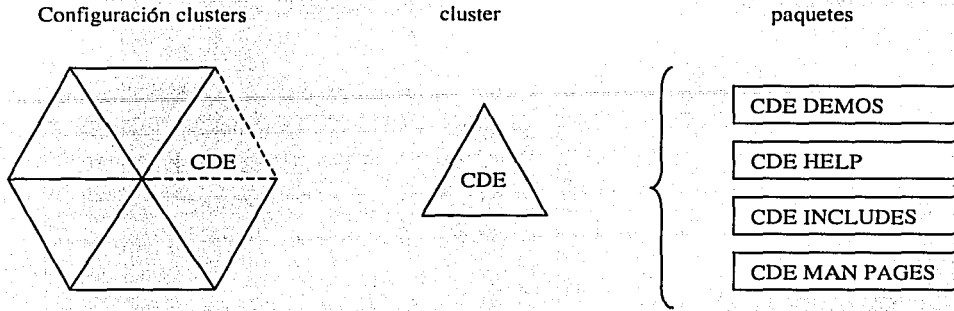


FIG. 3.6 Agrupación de los paquetes.

Paquetes:

Los paquetes son un grupo de software que contiene programas relacionados entre sí guardados en archivos y directorios. Por ejemplo, todos los archivos y directorios relacionados a las páginas del manual están en el paquete llamado SUNWman.

Los paquetes son administrados usando los comandos de administración y generalmente son identificados como SUNWXXX que es la convención adoptada para identificarlos.

Clusters:

Durante la instalación los paquetes son agrupados dentro de clusters, siendo estos colecciones lógicas de paquetes. Por ejemplo, el CDE clusters incluye los siguientes paquetes:

SUNWdtab	SUNWdthed	SUNWdtmad	SUNWeudhr
SUNWdtbas	SUNWdthev	SUNWdtrme	SUNWeudhs
SUNWdtbem	SUNWdticn	SUNWdtwn	SUNWeudis
SUNWdtmnn	SUNWdtim	SUNWeudba	SUNWeudlg
SUNWdtst	SUNWdtinc	SUNWeudbd	SUNWmfman
SUNWdthe	SUNWdtma	SUNWeudda	

Algunos clusters solo incluyen un paquete, por ejemplo, las páginas del manual están dentro de SUNWman.

Configuración del clusters

La instalación del sistema operativo esta dada por grupos de software que incluyen paquetes y clusters, estos grupos se dividen en cinco formas diferentes de instalar el S. O. Dependiendo de los requerimientos del usuario. Las cinco configuraciones del clusters son:

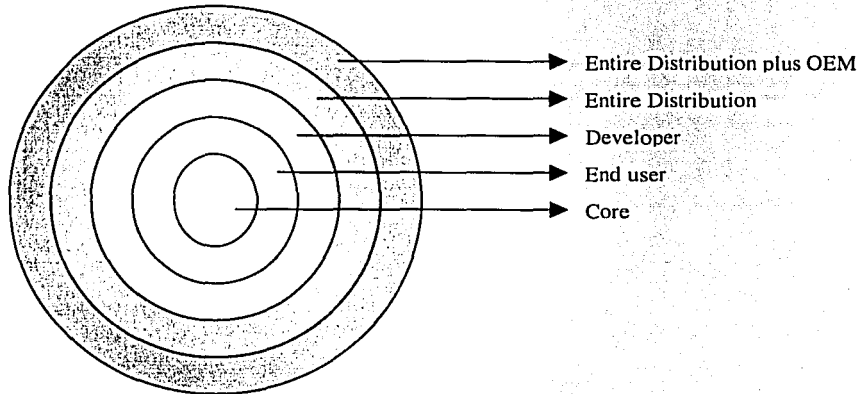


Fig. 3.7 Configuración del Clusters.

Los tamaños de la configuración son basados en el tipo de máquina y en el número de clientes que soportará el sistema. Durante la instalación del sistema operativo son recomendados los tamaños de la configuración del clusters basados en los parámetros anteriores.

Core:

Esta configuración contiene el software mínimo necesario para iniciar y correr el ambiente del sistema operativo Solaris 7, lo que permite ser usado para configurar sistemas standalone, pero no servidores. Dicha configuración incluye algunos paquetes para redes y manejadores necesarios para correr el OpenWindows y el CDE, pero no incluye el software de las páginas del manual.

End User:

Esta configuración contiene el software necesario para correr en el modo de simple usuario y correr el ambiente del sistema operativo Solaris 7; también incluye el OpenWindows Versión 3.x y el Common Desktop Environment, pero no las páginas del manual.

Developer:

Esta configuración contiene el software necesario para desarrollar software en el ambiente Solaris 7, incluye herramientas de compilación, el CDE y el OpenWindows, además de las páginas del manual.

Entire Distribution and Entire Distribution Plus OEM.

Estas dos configuraciones contienen todo el sistema operativo Solaris 7 con la diferencia de que la Entire Distribution Plus OEM (Original Equipment Manufacturer) tiene además controladores de hardware que le permiten un mejor manejo de los mismos.

Es importante mencionar que dependiendo del tipo de configuración de software que se seleccione es posible personalizarla, adicionando o eliminando clusters o paquetes.

Requerimientos de hardware

Para la instalación del sistema operativo utilizando la utilería SunInstall se necesita cumplir con los requerimientos mínimos para obtener una buena funcionalidad del equipo.

- El equipo debe estar basado en arquitectura SPARC o sistema Intel.
- Se debe tener por lo menos 1 Gbyte de espacio en disco para la instalación del sistema operativo.
- Se deben de tener por lo menos 64 Mbytes de memoria.
- Una unidad de CD-ROM o por lo menos acceso a una red para poder hacer la instalación.

Información para la identificación del sistema

Durante la primera parte del proceso de la instalación del sistema operativo, la herramienta SunInstall solicita el nombre de cómo se llamará el sistema e información básica para la red; es importante tener esta información reunida antes de empezar la instalación para que sea más rápida; los datos que se necesitan son los siguientes:

Host name: El host name es un nombre que se le proporciona a la máquina para identificarla dentro del sistema y de la red, cabe mencionar que puede identificarse por medio de la dirección IP; pero si se ubica dentro de una red donde se encuentran varios servidores o máquinas conectadas, será difícil recordar el número, para esto se creo el host name, para hacer más fácil la identificación del equipo.

IP address: Es un número con el cual se identifica a nuestro equipo dentro de una red interna o externa, es irrepetible y mediante él se puede localizar al equipo. Este número es asignado dependiendo de las necesidades de la red.

Name service (NIS, NIS+, Other or None): El servicio de nombres es un conjunto de opciones que permite seleccionar cuál va a ser el tipo de servicio de red.

Subnet (yes or no): Si el equipo formara parte de una subred es importante conocerla para conectarlo a ella.

Geographic location: Otro dato que solicita durante los primeros pasos de la instalación es la ubicación geográfica, ésta es seleccionada de una lista de opciones dada por el sistema.

Time zone information: También se preguntará la hora y la fecha para poner su reloj interno a tiempo.

Root password: Finalmente, al término de la instalación pedirá la contraseña para acceder como super usuario dentro del sistema.

Preparando la instalación

Para la instalación del sistema operativo por primera vez, se deben reunir los puntos mencionados en el tema anterior, para una actualización se llevarán algunos pasos adicionales que se describirán a continuación, aunque este trabajo se base únicamente en la instalación por primera vez.

- 1.- Accesar al sistema como root (también conocido como super usuario). Esto se debe realizar ya que es la manera de tener acceso a todos los sistemas de archivos y comandos, por lo tanto es el único modo de realizar la actualización.
- 2.- Avisar a los usuarios de la actualización para que respalden sus archivos, los guarden y salgan del sistema.
- 3.- Realizar el respaldo de todo el sistema por si fuera necesario cargarlo de nuevo.
- 4.- Dar de baja el equipo dejándolo en el nivel del ok.
- 5.- Insertar el CD-ROM dentro de la unidad.
- 6.- Darle la instrucción al sistema para que arranque desde el CD, ejemplo:

ok boot cdrom

Una vez empezado el programa de instalación, es cargado en la memoria RAM, los cambios no se graban al disco hasta que se termina el proceso de instalación.

En el tema siguiente se presentará una secuencia de las ventanas que aparecerán durante la instalación del sistema operativo Solaris 7.

Selección del lenguaje y la localidad

La primera ventana que aparecerá durante la instalación es la de selección del idioma y la localidad para que las siguientes pantallas se desplieguen en inglés u otro idioma disponible. (Fig. 3.8)

Una vez seleccionado el idioma (languages) y la localidad para el sitio, se presiona la tecla para continuar.

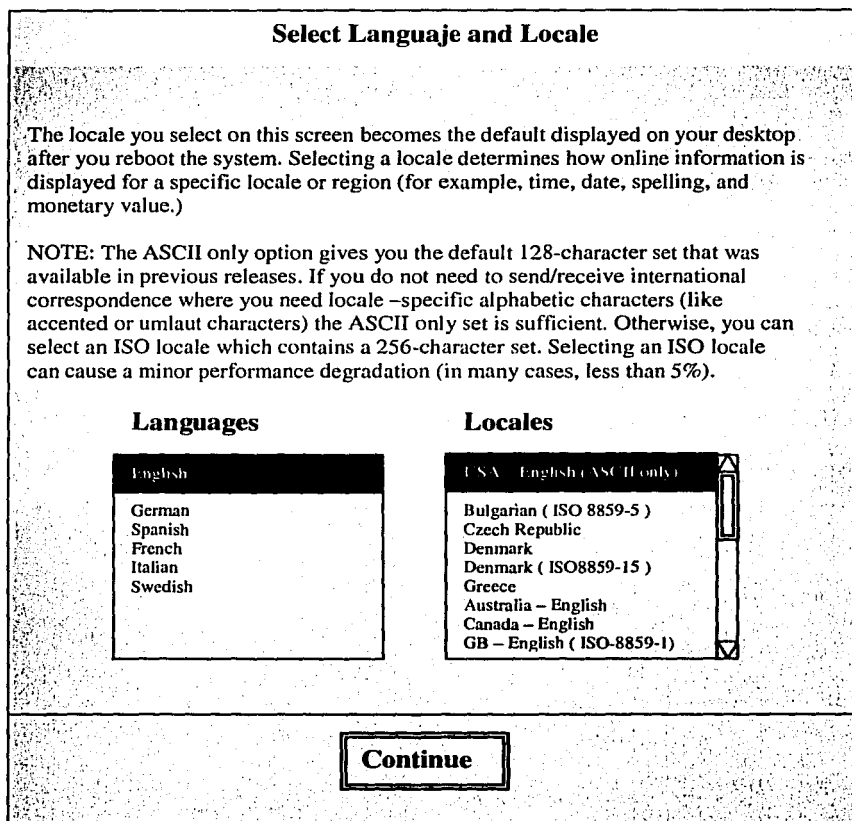


Fig. 3.8 Seleccionar el lenguaje y la localidad.

Instalación en sección

Esta ventana informa cómo se llevará a cabo el proceso de instalación. (Fig. 3.9)

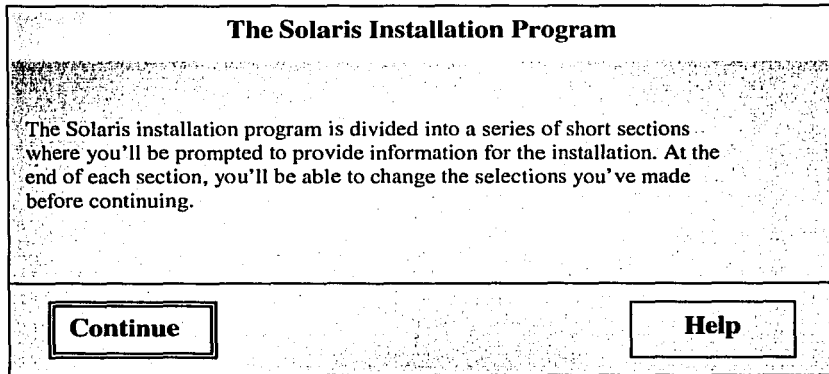


Fig. 3.9 Instalación del programa Solaris.

Sección de identificación

Lee la descripción del proceso de la identificación. Presionar el botón Help si se quiere más información en esta parte de la instalación, o presionar el botón de continue para seguir con la instalación. (Fig. 3.10)

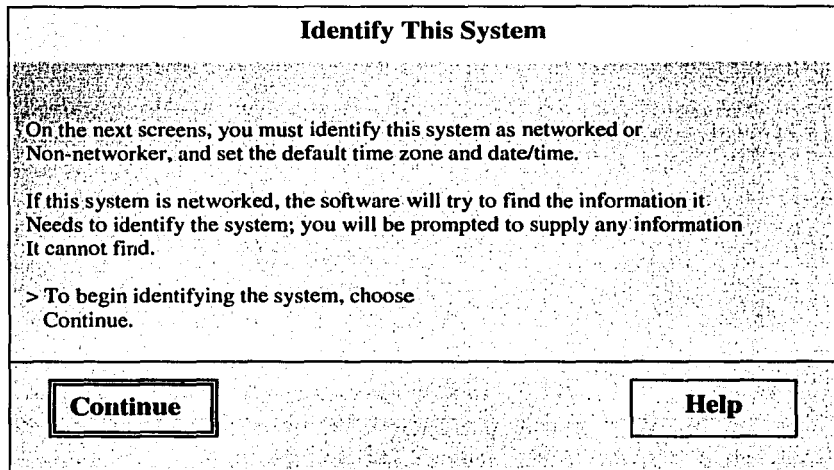
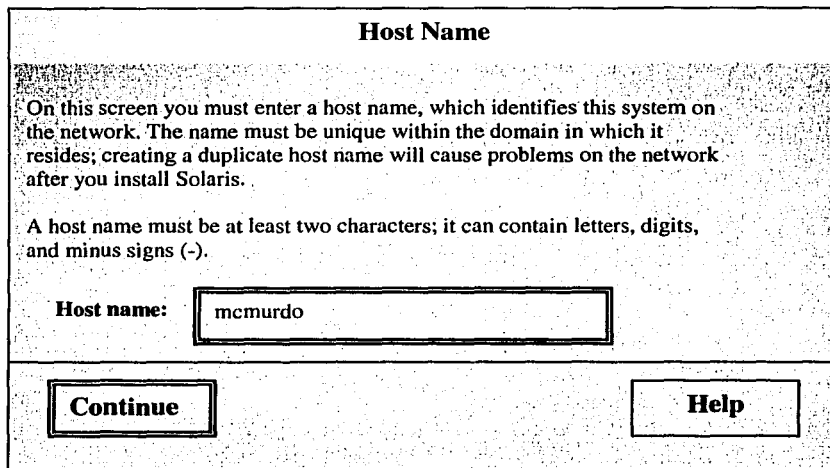


Fig. 3.10 Identificación del sistema.

Host Name

Hacer clic en el rectángulo para escribir el nombre del sistema y presionar continue para seguir adelante con la instalación. (Fig. 3.11)

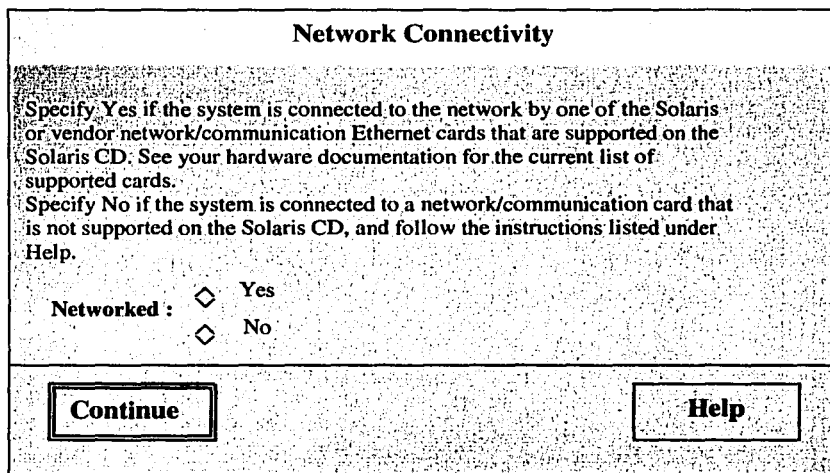


The screenshot shows a window titled "Host Name". The text inside reads: "On this screen you must enter a host name, which identifies this system on the network. The name must be unique within the domain in which it resides; creating a duplicate host name will cause problems on the network after you install Solaris." Below this, it says: "A host name must be at least two characters; it can contain letters, digits, and minus signs (-)." There is a text input field labeled "Host name:" containing the text "mcmurdo". At the bottom of the window, there are two buttons: "Continue" on the left and "Help" on the right.

Fig. 3.11 Host Name.

Conexión a red

Esta ventana confirmará que el equipo será conectado a una red. (Fig. 3.12)

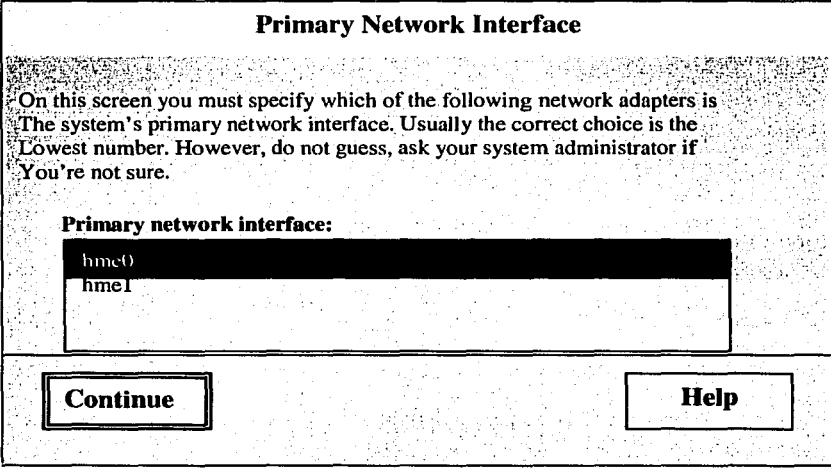


The screenshot shows a window titled "Network Connectivity". The text inside reads: "Specify Yes if the system is connected to the network by one of the Solaris or vendor network/communication Ethernet cards that are supported on the Solaris CD. See your hardware documentation for the current list of supported cards." Below this, it says: "Specify No if the system is connected to a network/communication card that is not supported on the Solaris CD, and follow the instructions listed under Help." There are two radio button options: "Networked : Yes" and " No". At the bottom of the window, there are two buttons: "Continue" on the left and "Help" on the right.

Fig.3.12 Conexión a red.

Interfaz primaria de red

Seleccionar la interfaz de red que se usará, esta opción solo se seleccionará si existe un adaptador de red instalado en el sistema. (Fig. 3.13)



Primary Network Interface

On this screen you must specify which of the following network adapters is The system's primary network interface. Usually the correct choice is the Lowest number. However, do not guess, ask your system administrator if You're not sure.

Primary network interface:

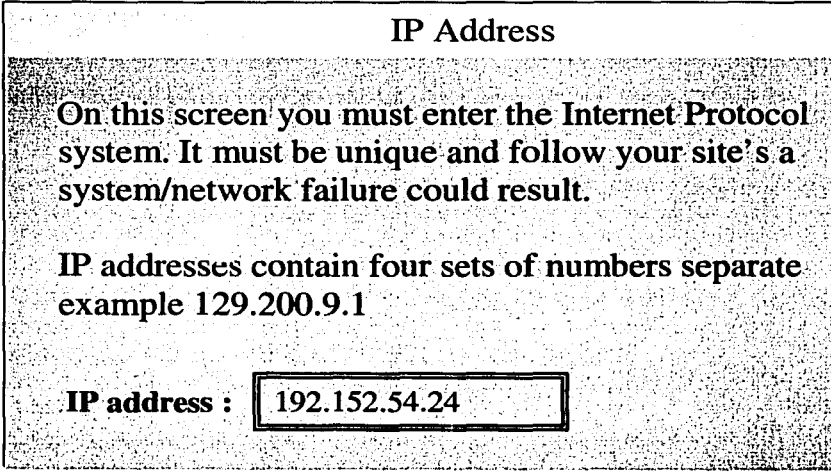
hmc0
hmc1

Continue **Help**

Fig. 3.13 Interfaz primaria de red.

Dirección IP

Se debe escribir la dirección IP del sistema dentro del cuadro de texto. (Fig. 3.14)



IP Address

On this screen you must enter the Internet Protocol system. It must be unique and follow your site's a system/network failure could result.

IP addresses contain four sets of numbers separate example 129.200.9.1

IP address :

Fig. 3.14 Dirección IP.

Confirmación de la información de identificación

En la ventana presentada por la figura 3.15 se deberá verificar que la información suministrada sea la correcta. Si se encuentra algún error se podrá corregir ya que todavía no es registrada por el sistema.

Confirm Information

> Confirm the following information. If it is correct, choose Continue; to change any information choose Change.

Host name:	mcmurdo
Networked:	Yes
Primary network interface:	hme0
IP address:	192.152.54.24

Continue

Change

Help

Fig. 3.15 Confirmación de la información de identificación.

Información de red

En esta ventana se seleccionará el servidor de nombres que usará el sistema: NIS+, NIS, otro como DNS, o ninguno. En esta instalación no se usará ninguno. (Fig. 3.16)

Name Service

On this screen you must provide name service information.

> Select NIS+ or NIS if this system is known to the name server; Select Other if your site is using another name service (for example, DCE or DNS); select None if your site is not using a name service, or if it is not yet established.

Name service:

- NIS+
- NIS (formerly yp)
- Other
- None

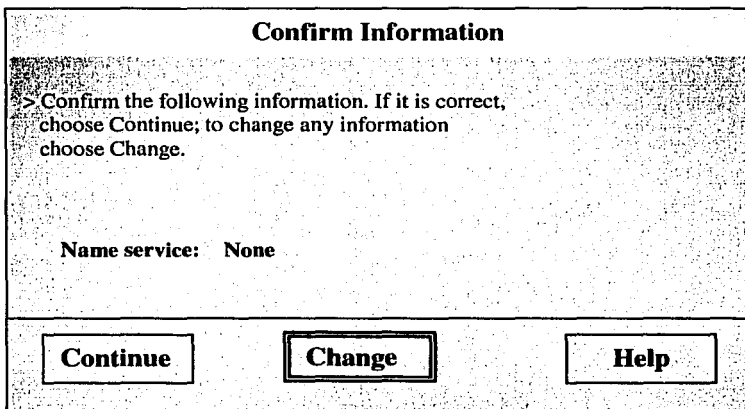
Continue

Help

Fig. 3.16 Servidor de nombres.

Confirmación del servidor de nombres

En esta ventana se muestra el servidor de nombre elegido en la ventana anterior para su confirmación, si se quiere cambiar la información dada se presiona el botón de cambio y regresará a la ventana anterior, de lo contrario se deberá oprimir continue. (Fig. 3.17)



Confirm Information

> Confirm the following information. If it is correct, choose Continue; to change any information choose Change.

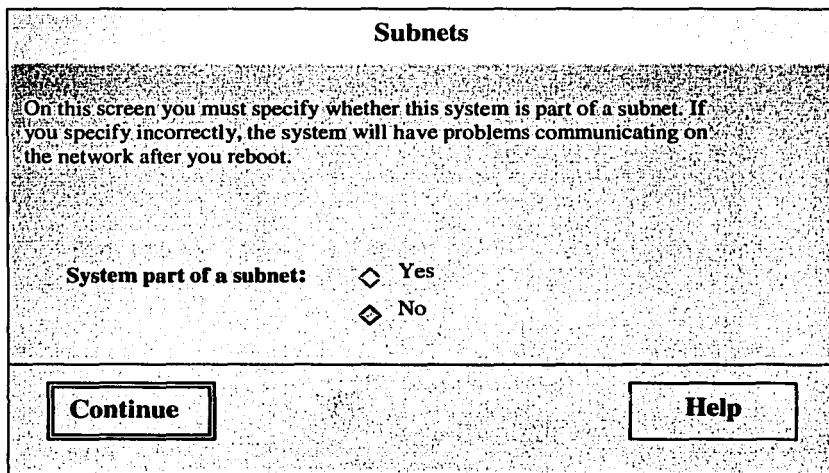
Name service: None

Continue Change Help

Fig. 3.17 Confirmación del servidor de nombres

Subred

Para muchas compañías es necesario tener varias subredes dentro de su red WAN, para este trabajo no se instalará una subred. Si se selecciona que sí pertenecerá a una subred, la siguiente ventana preguntará cuál es la máscara de la subred. (Fig. 3.18)



Subnets

On this screen you must specify whether this system is part of a subnet. If you specify incorrectly, the system will have problems communicating on the network after you reboot.

System part of a subnet: Yes
 No

Continue Help

Fig. 3.18 Ventana de subred.

Información del tiempo y zona

La configuración del tiempo y zona puede ser seleccionada de tres formas, la más usada es seleccionando en la ventana región la región geográfica correspondiente. (Fig. 3.19)

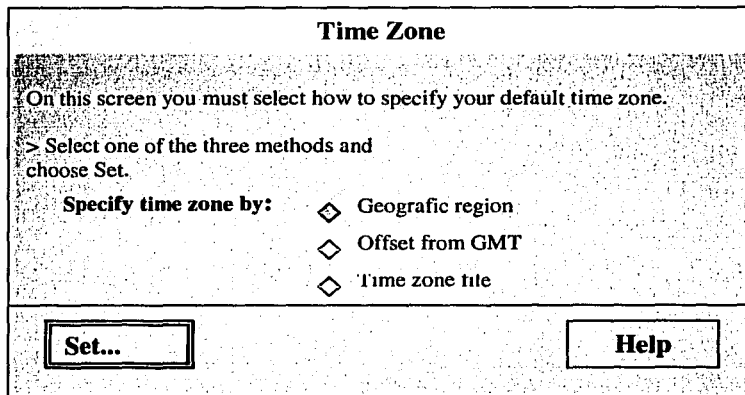


Fig. 3.19 Tiempo y zona

La región geográfica

Seleccionar de la ventana la región geográfica de la localidad y el tiempo según la zona correspondiente donde se localizará el equipo. (Fig. 3.20)

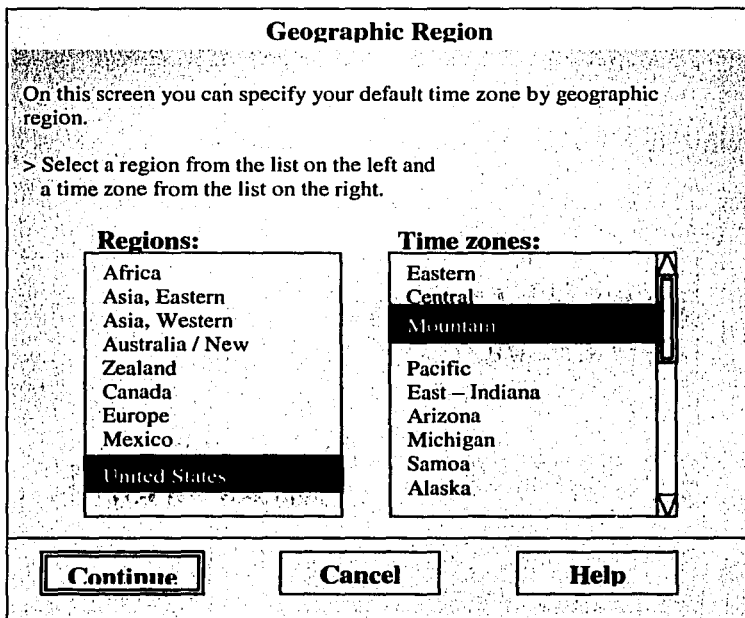


Fig. 3.20 Región geográfica.

Fecha y hora local

En esta ventana se debe verificar que la hora y fecha sean las correctas, en caso de que algún valor sea incorrecto se debe modificar dentro de los cuadros señalados y presionar el botón de continuar. (Fig. 3.21)

Date and time

> Accept the default date and time or enter new values.

Date and time : 1998-09-17 11:32

Year (4 digits) :

Month (1-12) :

Day (1-31) :

Hour (0-23) :

Minute (0-59s) :

Fig. 3.21 Fecha y hora.

Confirmación de la información de identificación

En esta ventana se debe verificar si es correcta la información para la red, región geográfica y la fecha y hora, pudiéndose hacer cambios en caso de haber algún error en la información. (Fig. 3.22)

Confirm Information

> Confirm the following information. If it is correct, choose Continue; to change any information choose Change.

System part of a subnet: No

Time zone: US/Mountain

Date and time: 1998-10-16 14:08:00

Fig. 3.22 Confirmación de la información.

Una vez llegado a este punto de la instalación aparecerá en la ventana de Consola un mensaje avisando que ha sido completada la identificación del sistema, seguido por un mensaje para iniciar el programa de instalación del sistema operativo.

En la siguiente ventana (Fig. 3.23), se deberá leer la información del recuadro y después oprimir el botón inicial para continuar con la instalación.

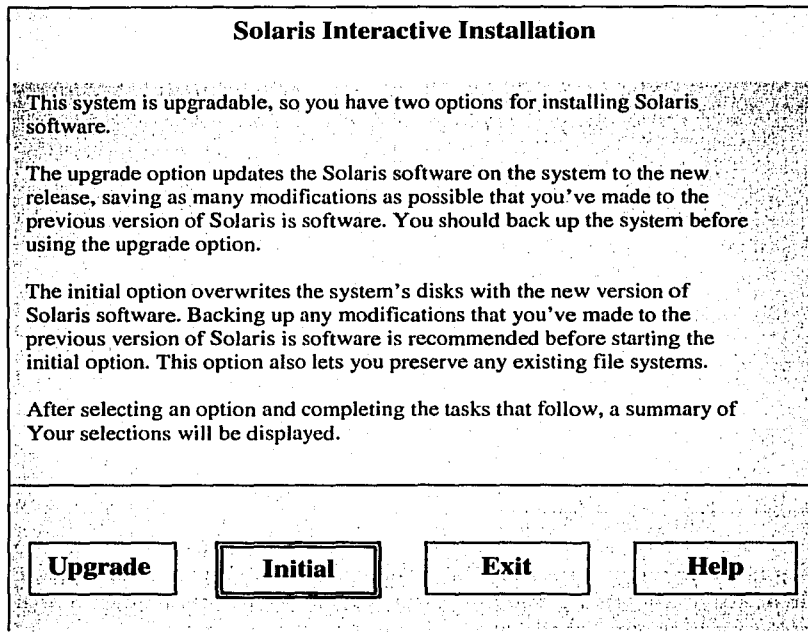


Fig. 3.23 Ventana inicial.

Instalación inicial

Cuando se empieza una instalación inicial del software, el sistema informa de una serie de eventos que serán llevados a cabo. (Fig. 3.24)

Se debe leer la información desplegada en la ventana de la figura siguiente y presionar continue para proceder con el proceso de instalación o exit para terminarlo.

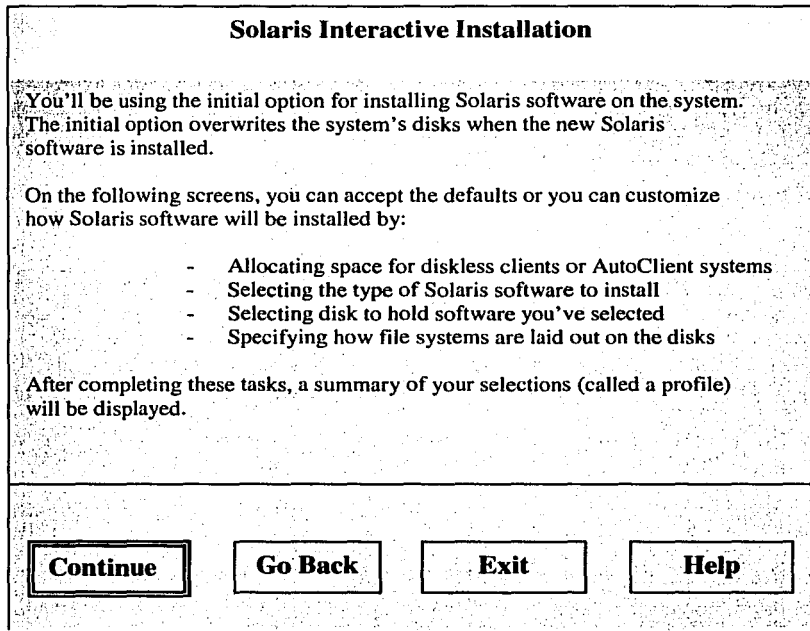


Fig. 3.24 Instalación interactiva de Solaris.

Identificando el tipo de sistema

La pantalla siguiente pregunta el proceso de instalación, si deseamos reservar un espacio en disco para conectar al sistema alguna estación de trabajo sin disco propio (Fig. 3.25). Este proceso de instalación se hará para una estación de trabajo standalone.

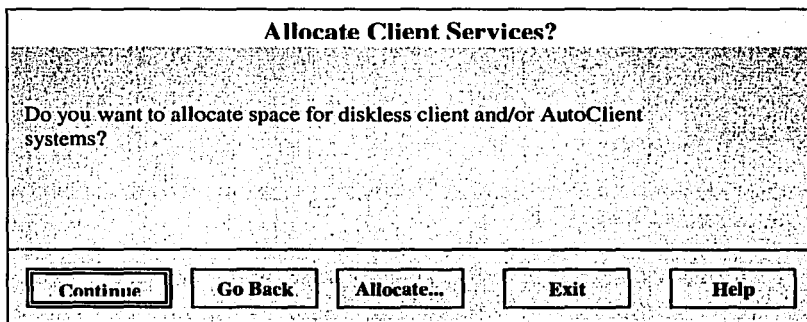


Fig. 3.25 Alojamiento de clientes.

Selección de lenguaje adicional

En esta ventana se puede seleccionar un idioma adicional para que los mensajes siguientes aparezcan en un idioma opcional. Se tiene que seleccionar el idioma y después adicionarlo con el botón. (Fig. 3.26)

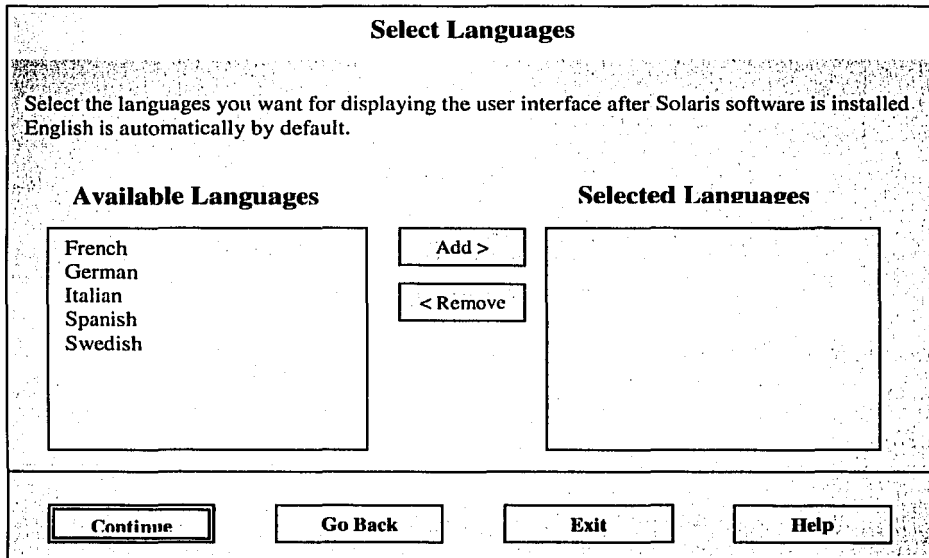


Fig. 3.26 Selección del lenguaje de las ventanas.

Tipo de instalación

Es posible seleccionar el grupo de software que se necesitará para los requerimientos de la estación de trabajo, en la figura 3.27 se pueden ver los tamaños necesarios para cada tipo de configuración del sistema, éstos pueden variar dependiendo del tipo de sistema y arquitectura del kernel que contenga el equipo.

La pantalla seleccionará por default la configuración de Entire Distribution plus OEM.

Selección de discos

En esta imagen se muestran, en la lista del lado izquierdo, los manejadores de disco que son conocidos por el sistema, estos discos fueron seleccionados durante la instalación inicial. También esta lista muestra el disco de inicio seleccionado automáticamente. (Fig. 3.29)

Select Disks

Select the disks for installing Solaris software. Start by looking at the Required field; this value is the approximate space needed to install the software you've selected. Keep selecting disks until the Total Selected value exceeds the Required value.

> To move a disk from the Available to the Selected window, click on the disk, then click on the > button.

Available Disks		Selected Disks
c1t3d0 4092 MB	> < >> <<	c0t0d0 (boot disk) 4102 MB
Total Available: 4092		Recommended: 1087
Boot Device: c0t0d0s0		Required: 826
Select Root Location		Total Selected: 4102

Continue
Go Back
Exit
Help

Fig. 3.29 Selección de dos disco.

Guardando datos existentes

Después de seleccionar los manejadores de disco, si es que existe alguna partición que se quiera guardar, presionar el botón preserve para que despliegue una nueva ventana donde se seleccione la partición. Si se está preparando una nueva instalación basta con presionar continue. (Fig. 3.30)

Preserve Data?

Do you want to preserve existing data? At least one of the disks you've selected for installing Solaris software has file systems or unnamed slices that you may want to save.

Continue
Go Back
Preserve..
Exit
Help

Fig. 3.30 Ventana para guardar una partición existente.

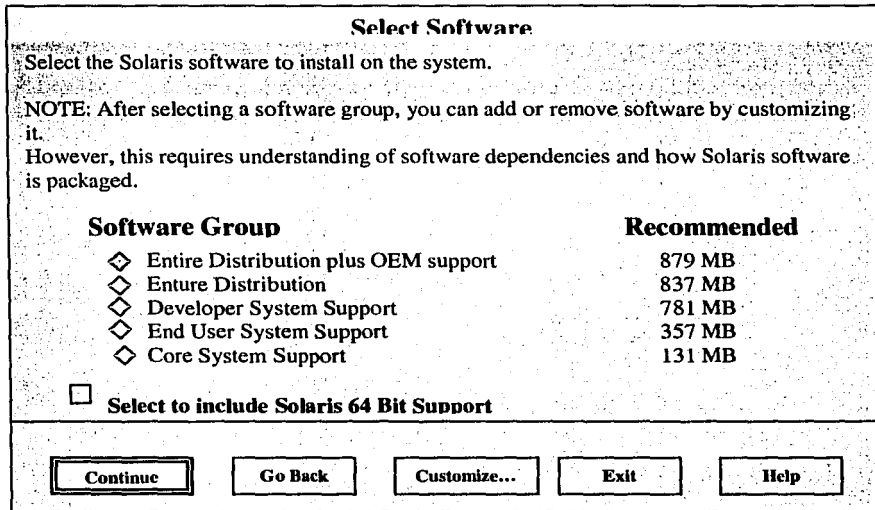


Fig. 3.27 Selección del Software.

Seleccionado los discos de instalación

En esta ventana solamente se seleccionarán los discos para la instalación del software de Solaris. (Fig. 3.28)

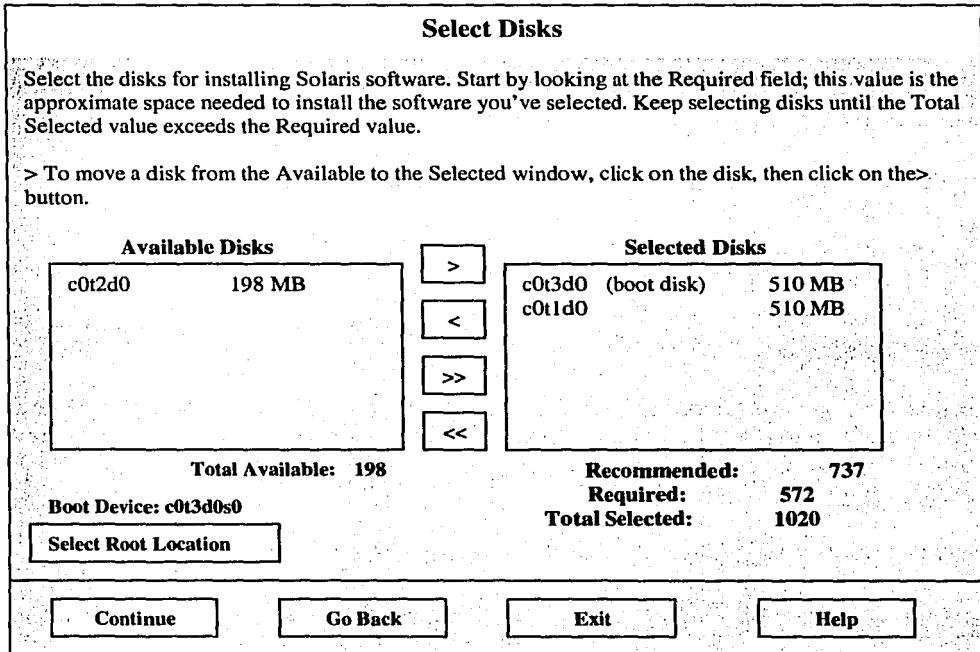


Fig. 3.28. Selección de discos.

Selección de discos

En esta imagen se muestran, en la lista del lado izquierdo, los manejadores de disco que son conocidos por el sistema, estos discos fueron seleccionados durante la instalación inicial. También esta lista muestra el disco de inicio seleccionado automáticamente. (Fig. 3.29)

Select Disks

Select the disks for installing Solaris software. Start by looking at the Required field; this value is the approximate space needed to install the software you've selected. Keep selecting disks until the Total Selected value exceeds the Required value.

> To move a disk from the Available to the Selected window, click on the disk, then click on the > button.

Available Disks		Selected Disks
c1t3d0 4092 MB	>	c0t0d0 (boot disk) 4102 MB
	<	
	>>	
	<<	
Total Available: 4092		Recommended: 1087
Boot Device: c0t0d0s0		Required: 826
Select Root Location		Total Selected: 4102

Continue
Go Back
Exit
Help

Fig. 3.29 Selección de dos disco.

Guardando datos existentes

Después de seleccionar los manejadores de disco, si es que existe alguna partición que se quiera guardar, presionar el botón preserve para que despliegue una nueva ventana donde se seleccione la partición. Si se está preparando una nueva instalación basta con presionar continue. (Fig. 3.30)

Preserve Data?

Do you want to preserve existing data? At least one of the disks you've selected for installing Solaris software has file systems or unnamed slices that you may want to save.

Continue
Go Back
Preserve..
Exit
Help

Fig. 3.30 Ventana para guardar una partición existente.

Partición del disco

El botón de Manual Layout permite crear los tamaños de las particiones del disco manualmente, Auto Layout determina los tamaños para las particiones del disco automáticamente.

Una ventaja de oprimir la tecla Auto Layout es que despliega la instalación del tamaño de las particiones por default. Adicionalmente, se pueden modificar de nuevo los tamaños de las particiones después de que se han colocado los tamaños automáticamente. (Fig. 3.31)

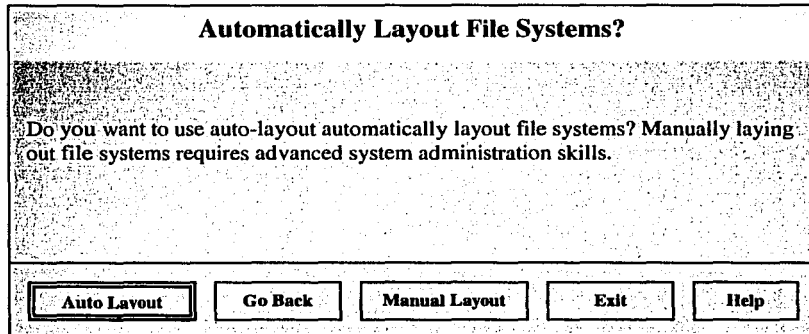


Fig. 3.31 Particionando el disco.

File systems

Una vez aceptada la partición manual de los discos aparecerá la siguiente ventana.

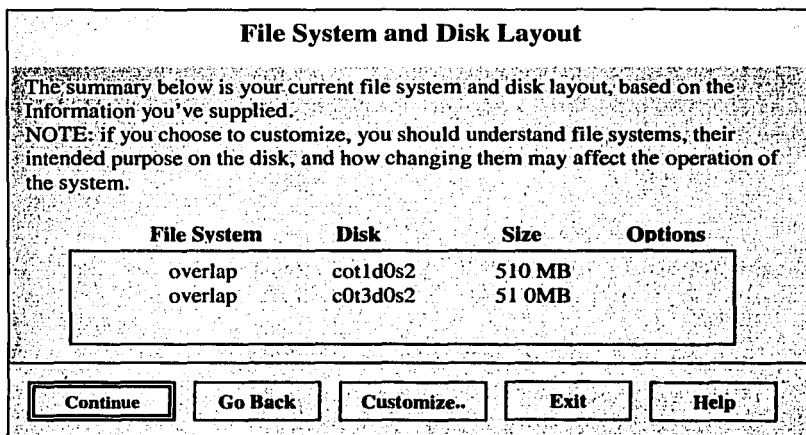


Fig.3.32 Partición manual de varios discos.

En esta ventana se muestra la partición manual de un solo disco de 4GB.

Customize Disks

/

Recommended **Minimum**

78 **67**

	Disk: c0t0d0 4102 MB	
0	/	1000
1	swap	64
2	overlap	4102
3	/export	50
4	/export/swap	50
5		
6	/usr	2937
7		
	Capacity: 4102MB	
	Allocated: 4102MB	
	Free: 0 MB	

Boot Device: c0t0d0s0

Fig. 3.35 Ejemplo de partición de un disco de 4GB.

Las siguientes ventanas (Figs. 3.36 y 3.37) muestran las particiones introducidas en las ventanas anteriores mostrando los tamaños definidos.

File System and Disk Layout

The summary below is your current file system and disk layout, based on the information you've supplied.

NOTE: if you choose to customize, you should understand file systems, their intended purpose on the disk, and how changing them may affect the operation of the system.

File System	Disk	Size	Options
overlap	c0t1d0s2	510 MB	
/var	c0t1d0s5	100 MB	
/usr/openwin	c0t1d0s6	410 MB	
/	c0t3d0s0	145 MB	
swap	c0t3d0s1	65 MB	

Fig. 3.36 Resultado de las particiones hechas a dos discos de 510 MB.

File System and Disk Layout

The summary below is your current file system and disk layout, based on the information you've supplied.

NOTE: if you choose to customize, you should understand file systems, their intended purpose on the disk, and how changing them may affect the operation of the system.

File System	Disk	Size	Options
/	c0t0d0s0	1000 MB	
swap	c0t0d0s1	64 MB	
overlap	c0t1d0s2	4102 MB	
/export	c0t0d0s3	50 MB	
/export/swap	c0t3d0s4	50 MB	

Fig. 3.37 Resultado de las particiones hechas a un disco de 4GB.

File system remoto

La ventana de la figura 3.38 presenta la posibilidad de montar un file system desde otro equipo, para esta instalación no se hará de esta manera, para seguir con la instalación basta con presionar el botón continue.

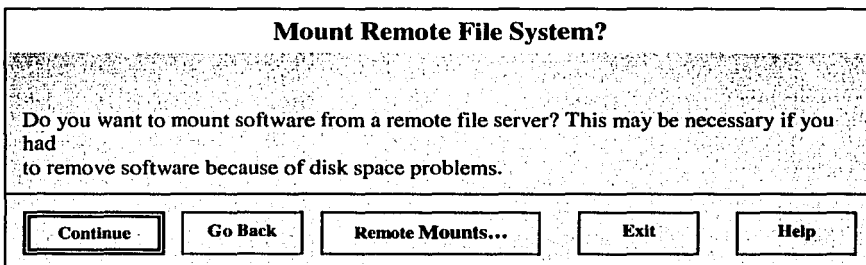


Fig. 3.38 File system remoto

Profile

La ventana profile (Figs. 3.39 y 3.40), es de configuración final antes de que el programa de instalación empiece. Se debe verificar la información presentada antes de presionar el botón para empezar la instalación. Una vez presionado el botón de Begin Installation el programa SunInstall empezará a leer CD-ROM para instalar el nuevo sistema operativo Solaris.

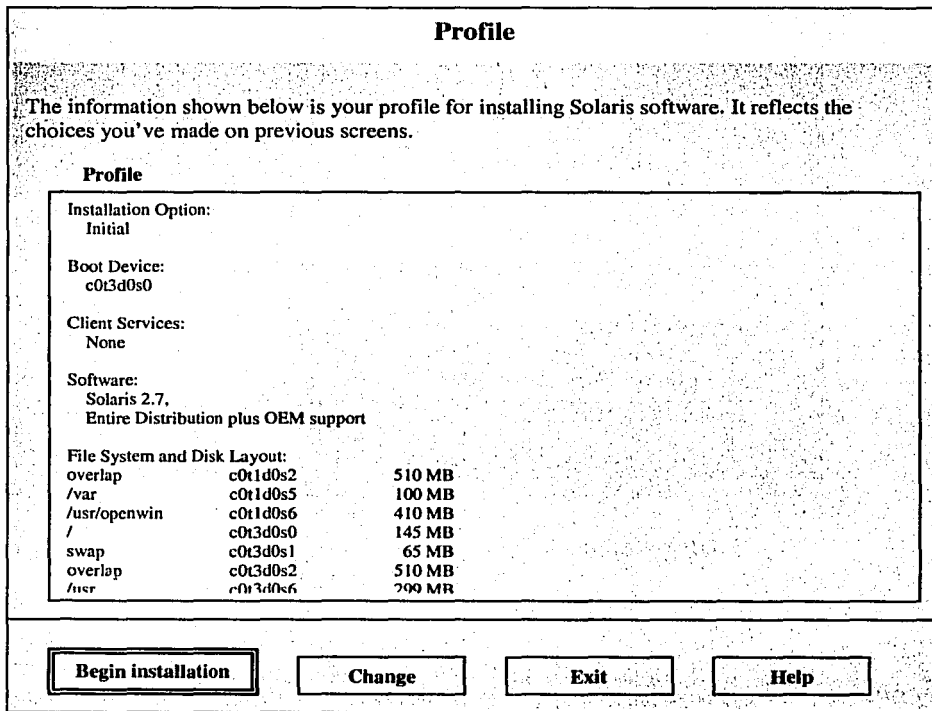


Fig. 3.39 Instalación en dos discos de 510 Mb.

ESTA TEXTO NO SALE DE LA BIBLIOTECA

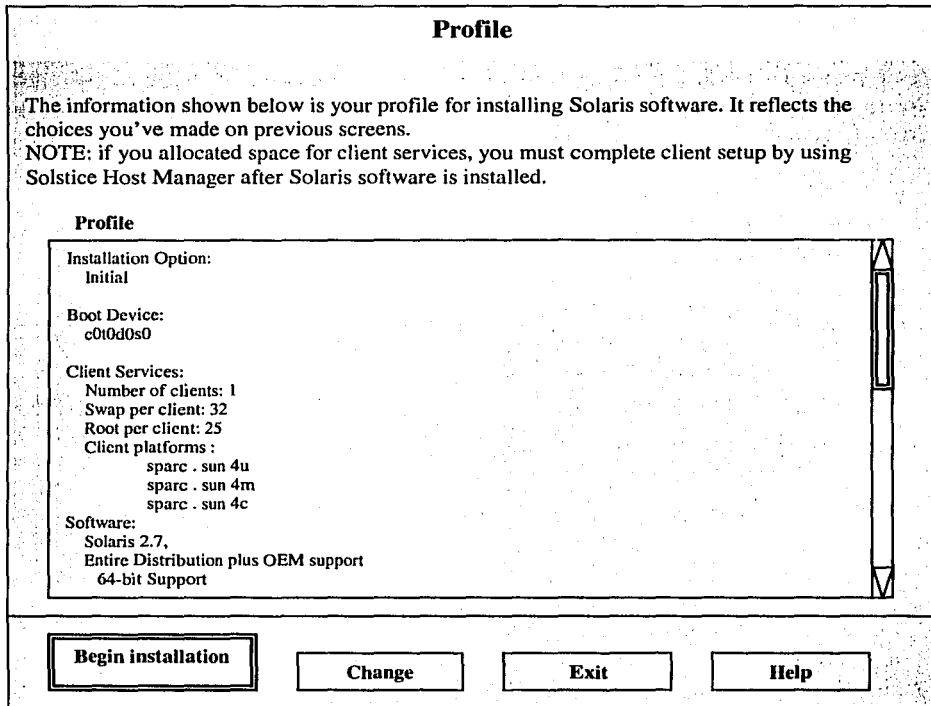


Fig. 3.40 Instalación en un disco de 4 Gb.

Reboot

Después de indicarle al programa que empiece con la instalación, una pequeña ventana es desplegada, presentando la opción de inicializar el sistema de forma automática o manual después de la instalación del sistema Solaris. (Fig. 3.41)

Se sugiere presionar el botón Auto Reboot.

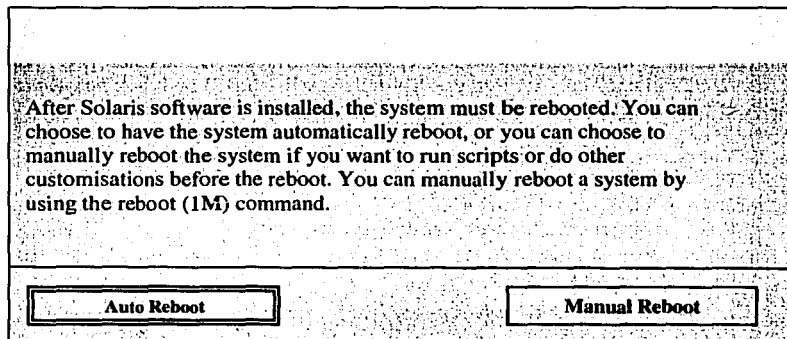


Fig. 3.41 Completar la instalación inicializando el sistema.

Ventana de progreso

Después de presionar la tecla de Auto Reboot aparecerá la ventana final (Fig. 3.42) que desplegará el avance durante el proceso de instalación del sistema operativo. Algunos mensajes serán colocados al principio de la ventana.

Cuando la instalación es completada, el sistema automáticamente inicializará si fue seleccionada la opción de la imagen anterior.

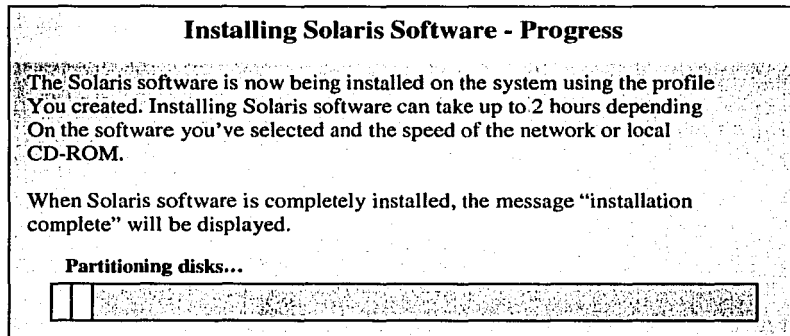


Fig. 3.42 Progreso de instalación.

Root password

Después de inicializar el sistema, aparecerá una ventana donde se pide que se inserte el password de Root. (Fig. 3.43)

Root es el superusuario del sistema requerido para el mismo. Este punto termina la instalación del sistema con el mensaje "The system is ready" y deja en una ventana con el promp esperando alguna orden.

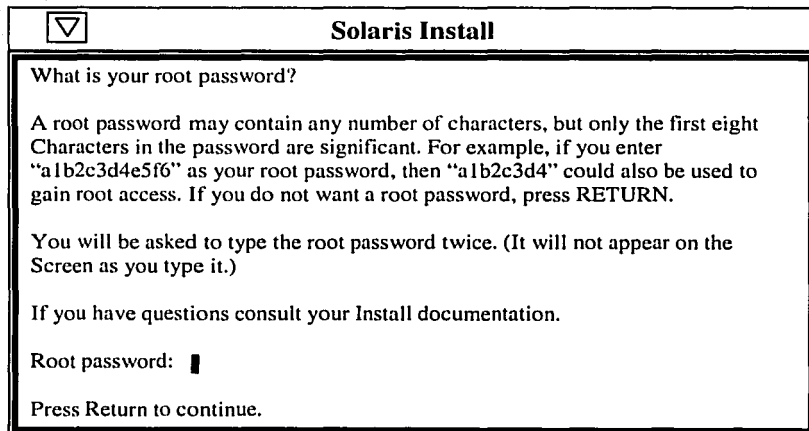


Fig. 3.43 Password de Root.

3.8 Administración

A todo el conjunto de labores que realiza el usuario de una computadora para que ésta funcione correctamente se le da el nombre de administración del sistema.

El propietario de una computadora debe estar al tanto de las tareas de administración del sistema, tomando en cuenta el tipo de sistema operativo con que se trabaje, el tamaño del sistema en general, si se encuentran conectados a él usuarios que trabajen con el sistema o con alguna aplicación en particular.

En algún momento se puede necesitar instalar software para el sistema, por lo que es necesario conocer los programas que se usarán y los dispositivos de hardware, como impresoras y unidades de disco, también se necesitará eliminar archivos innecesarios, optimizar discos y hacer copias de seguridad de los datos. Se pueden llevar a cabo varias tareas que le permitan trabajar de una forma eficaz o ampliar los alcances de su trabajo.

La administración del sistema puede ser dividida en básica y avanzada, dependiendo de las tareas que se realizarán.

No existe un criterio definido para poder dividir las tareas a realizar para la administración de un sistema operativo y poder decir que se realiza una administración básica o avanzada. Para la realización de este capítulo de tesis realicé personalmente la división de la administración, tomando como básica todas aquellas tareas que por experiencia son comúnmente usadas en este sistema dentro de los equipos SUN; la administración avanzada no es mencionada en este trabajo dejándolo abierto para futuras investigaciones sobre este tema.

En este capítulo se describen algunos temas importantes que se deben de tomar en cuenta para llevar a cabo una buena administración básica del sistema. Es importante mencionar que el nivel de la administración dependerá del tamaño y características del equipo, pero los conceptos que serán presentados pueden llevarse a cualquier equipo que cuente con este tipo de sistema operativo.

Cabe mencionar que un buen administrador no es aquel que se aprende todos los comandos o manuales del sistema, sino el que se forma con la experiencia y el buen uso de los recursos proporcionados por el equipo para dar soluciones rápidas y/o eficientes a las necesidades del mismo.

3.9 Encendido del equipo

El encendido del equipo es el primer paso para administrar el sistema, aunque es un proceso sencillo es importante seguirlo cuidadosamente, pues este procedimiento es diferente al que se lleva a cabo en una PC. Por ello se verán algunos de los principales aspectos para un adecuado manejo del equipo.

Para encender un equipo que se encuentra totalmente apagado se deben realizar algunos pasos importantes:

- Verificar que se encuentren conectados los dispositivos externos al servidor.
- Verificar que se encuentre conectado a la energía eléctrica y, de ser posible, que el equipo se encuentre respaldado por una fuente de poder ininterrumpida.
- Encender los dispositivos externos y esperar unos minutos para cerciorarse que se ejecute su proceso de inicio.
- Encender el servidor.

Una vez que se prende el servidor, éste empieza con una serie de pasos realizados por el Boot PROM antes de cargar el sistema operativo (Fig. 3.44). Si no se detectan errores y la variable de auto-boot está definida en el PROM como verdadera, éste será cargado automáticamente, de lo contrario el proceso de inicialización se detendrá en el prom ok y será necesario darle el comando "boot" para que cargue el sistema operativo.

Una vez cargado el sistema operativo, preguntará el nombre de usuario y el password como un medio de seguridad para preveer que otras personas hagan uso del equipo con mayores privilegios de los que pueda tener. Finalmente, una vez que han sido dados el nombre de usuario y password, el sistema dejará en una ventana llamada console donde se pueda empezar a trabajar con el servidor.

3.10 Pasos del Boot PROM

Al encender el equipo, el Boot PROM lleva a cabo una serie de pasos automáticos de diagnóstico y prueba del equipo para verificar que no exista algún problema.

1.- Se corren pruebas de diagnóstico.

El programa de boot PROM corre rutinas de diagnóstico para verificar el sistema de hardware y memoria.

2.- Es desplegado el banner identificador del sistema.

El tipo de modelo del equipo, el tipo de teclado, el número de revisión del PROM, el tamaño de la memoria RAM, el número serial del PROM, la dirección Ethernet y el host ID son desplegados.

3.- Es leída la etiqueta del disco localizada en el sector 0 en el dispositivo por default de arranque.

4.- Es buscado el programa de arranque que se encuentra programado en el PROM.

5.- Es cargado el programa de inicio.

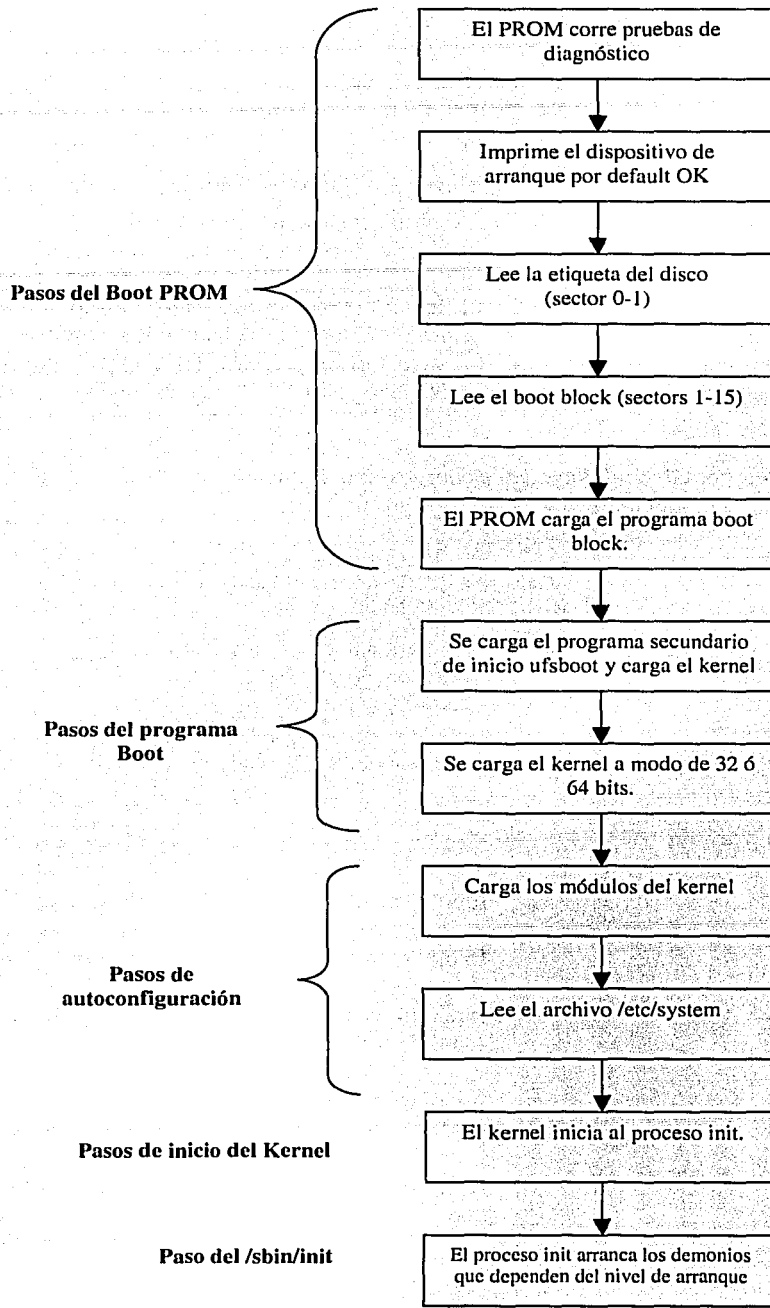


Fig. 3.44 Fases del sistema de arranque.

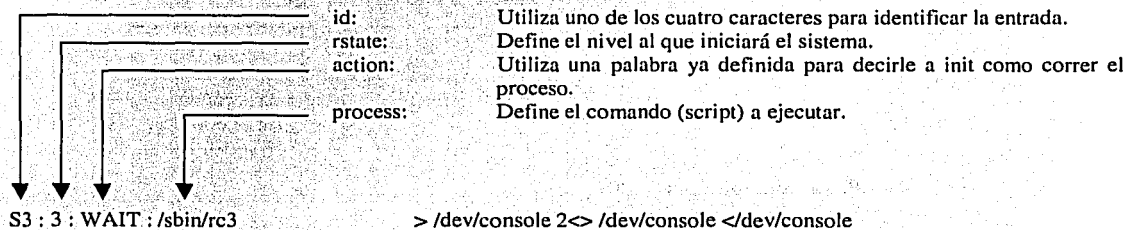
El archivo /etc/inittab

El archivo /etc/inittab contiene información que le proporciona al proceso init para indicarle los procesos que debe arrancar para el nivel al que debe iniciar el sistema y de cómo debe llevarlo a cabo, este archivo también le indica al proceso init tres funciones importantes:

- 1.- El nivel por default al que debe de iniciar.
- 2.- Qué acciones debe tomar cuando el sistema entra a un nuevo nivel.
- 3.- Qué procesos debe empezar, cuales deben existir y reiniciar si estos mueren.

Cuando el sistema inicia el proceso init lee el archivo inittab, busca el nivel por default que debe iniciar leyendo la línea del archivo initdefault, después ejecuta los comandos o scripts para las entradas que tiene la línea sysinit y ejecuta los scripts para las entradas que tienen 3 en la línea restante. Una vez que el proceso init realiza estos pasos el sistema inicia.

El archivo /etc/inittab contiene cuatro campos importantes:



Palabras clave

initdefault	Identifica el nivel por default al que inicializará el sistema.
respawn	Inicia los procesos y los restaura cuando éstos mueren.
powerfail	Inicia procesos definidos cuando el proceso init recibe una señal de fallo de la fuente de poder.
sysinit	Inicia los procesos antes de tratar de acceder a la consola y espera hasta que sean completados antes de continuar.
wait	Inicia los procesos y espera hasta que terminen antes de continuar con la siguiente instrucción para arrancar el nivel.

Existen más palabras claves que se encuentran en las páginas del manual relacionadas con el archivo inittab. El archivo inittab por default en el sistema se observa de la siguiente manera:

```
ap :: sysinit: /sbin/autopush -f/etc/u.ap
ap :: sysinit: /sbin/soconfig -f/etc/sock2path
fs :: sysinit: /sbin/rcS      sysinit>/dev/console 2<>/dev/console
</dev/console
is :3: initdefault:
p3 :s1234:powerfail : /usr/sbin/shutdown -y -i5 -
g0 >/dev/console 2<>/dev/console
sS :s : wait: /sbin/rcS      >/dev/console 2<>/dev/console </dev/console
s0 :0 : wait: /sbin/rc0      >/dev/console 2<>/dev/console </dev/console
s1 :1 : respawn: /sbin/rc1   >/dev/console 2<>/dev/console </dev/console
s2 :23 : wait: /sbin/rc2     >/dev/console 2<>/dev/console </dev/console
s3 :3 : wait: /sbin/rc3     >/dev/console 2<>/dev/console </dev/console
s5 :5 : wait: /sbin/rc5     >/dev/console 2<>/dev/console </dev/console
s6 :s : wait: /sbin/rc6     >/dev/console 2<>/dev/console </dev/console
fw :0 : wait: /sbin/uadmin 2 0 >/dev/console 2<>/dev/console </dev/console
of :5 : wait: /sbin/uadmin 2 6 >/dev/console 2<>/dev/console </dev/console
```

Un archivo importante de sistema es el directorio /sbin. El comando init lee uno de los 7 scripts (run control) especificados en este directorio (Fig. 3.45), que le indican las instrucciones que debe llevar a cabo para iniciar en el estado deseado. Los scripts son llamados rcx, donde la x significa el nivel.

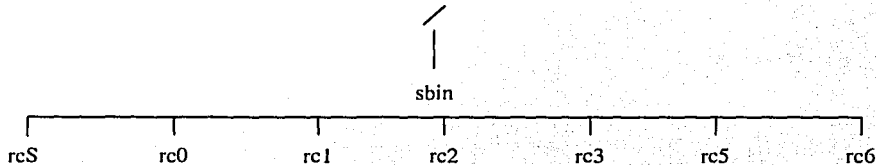


Fig. 3.45 Directorio /sbin

Los rc localizados en el directorio /sbin dirigen el sistema a leer los archivos localizados en el directorio /etc/rc#.d, por ejemplo, el script del directorio /sbin/rc2 leerá el archivo localizado en el directorio /etc/rc2.d. Estos directorios contienen los programas que inician o detienen los servicios del sistema. El directorio /etc contiene la misma estructura de los run control scripts que el directorio /sbin sólo que estos scripts son ligas a los scripts en el directorio /sbin.

Otro archivo importante de sistema es el directorio /etc/init.d, que contiene programas ligados al directorio /etc/rc*.d para iniciar o detener procesos de una forma apropiada.

Una ventaja importante de tener este directorio es que se puede iniciar o detener servicios individuales o procesos de una forma simple sin necesidad de cambiar de estado, por ejemplo, se puede detener o iniciar el servicio de lp con el siguiente comando:

```
# /etc/init.d/lp stop
# /etc/init.d/lp start
```


Arranque y desconexión

El sistema Solaris tiene diferentes modos de operación llamados estados del sistema, que hacen posible que el administrador pueda limitar la actividad del sistema cuando efectúe ciertas tareas de administración.

Por ejemplo, si está añadiendo una tarjeta de comunicaciones a la computadora, tendría que cambiar al estado de sistema 0 (estado de desconexión) y el sistema sería desactivado; si quiere ejecutar diagnósticos de hardware, puede cambiar al estado 5 y el sistema operativo se detendrá, pero se podrán ejecutar pruebas de diagnóstico.

Los dos tipos de estados del sistema de ejecución son los estados monousuario y multiusuario. Cuando se activa el sistema en estado monousuario sólo es accesible (está montado) el sistema de archivos de raíz (/) y sólo puede acceder a la computadora desde la consola. Cuando se activa el sistema en estado multiusuario, generalmente se montan todos los otros sistemas de archivos de la computadora y se inician procesos para permitir que se presenten los usuarios generales.

Por default, el sistema irá al estado multiusuario al arrancar. En general, al pasar a estados de sistema de número superior se inician procesos y montan sistemas de archivos, haciendo disponibles más servicios. Al pasar a estados de sistema de número inferior, al contrario, se tiende a hacer disponible un menor número de servicios.

El comando init

El comando init puede ser usado para cambiar el estado del sistema. Para utilizarlo se debe saber a que estado se desea pasar y ser superusuario del sistema.

Formato del comando

init [012356abcQqSs]

Opciones

- 0 Pasa el estado del sistema al nivel de PROM. En este estado la máquina está desconectada, en un punto en el que puede reinicializarse o apagarse.
- 1 Pasa al estado de simple usuario donde algunos archivos de sistema son montados y el acceso de usuarios es deshabilitado. En este estado los sistemas de archivos están disponibles, pero los procesos multiusuario, como los que permiten a los usuarios presentarse desde terminales distintos, no están disponibles.
- 2 Pasa al estado multiusuario sin compartir los recursos.
- 3 Pasa al estado multiusuario compartiendo los recursos.
- 4 Este es un estado definido por el usuario, no por el sistema.
- 5 Interrumpe el sistema y se apaga totalmente el equipo.
- 6 Estado de parada y reinicio. Detiene el sistema operativo y luego reinicia en el estado definido.

- a, b, c Pseudoestados. Procesa las entradas del archivo `/etc/inittab` que tienen asignadas el estado de sistema a, b ó c. Se trata de pseudoestados que pueden ser usados para ejecutar ciertas órdenes sin cambiar el estado de sistema actual.
- Q, q Reexamina el archivo `/etc/inittab` para determinar el nivel de ejecución actual.
- S, s Pasa el sistema a nivel de simple usuario. Todos los sistemas de archivos multiusuario están desmontados, los procesos multiusuario están eliminados y el sistema solamente puede ser accedido a través del teclado.

3.12 Apagado

El apagado del sistema es un proceso relativamente sencillo solo hay que seguir algunos pasos

1. Avisar a los usuarios que se dará de baja el sistema para que tengan tiempo de guardar su información y salir del sistema sin problemas.
2. Dar de baja las aplicaciones que estén corriendo en el sistema.
3. Aplicar al sistema un comando para dar de baja el equipo como el `init` con la opción 0 o el comando `shutdown` o el comando `halt`.
4. Apagar el sistema desde su interruptor junto con los dispositivos externos.

El comando `shutdown`

El comando `/usr/sbin/shutdown` es usado para cambiar el nivel al que corre el sistema. El beneficio de usar este comando en lugar de `init` es que con `shutdown` se puede notificar a los usuarios del apagado del equipo.

Si el comando `shutdown` es usado para pasar el sistema al nivel 0, el sistema operativo es dado de baja limpiamente, es decir, que todos los procesos son detenidos y los sistemas de archivos son desmontados.

Formato del comando

Shutdown [-y] [-g segundos] [-i nivel de inicio] [mensaje]

Opciones

- y Yes, le indica al sistema que no pida confirmación antes de desconectarse.
- g Periodo de gracia, con esta opción se le puede indicar al sistema que se desconecte en un tiempo determinado en segundos por el administrador antes de que se desconecte.
- i Permíte indicarle al sistema a que estado se desea que pase.
- Mensaje Se puede adicionar un mensaje a la cadena `shutdown` para informarle a los usuarios que se dará de baja el equipo, este mensaje deberá estar encerrado entre comillas.

3.13 Archivos de administración

Dentro del sistema operativo se encuentran agrupados sistemas de archivos que son importantes para el equipo y para los usuarios, por lo tanto, también son de importancia para el administrador del sistema, ya que en ellos se encuentra información que podría ser utilizada en muchos casos.

Los sistemas de archivos sobre los que el administrador deberá tener conocimiento se encuentran en: raíz (/), /var, /usr, /home.

El sistema de archivos raíz

El sistema de archivos raíz contiene todos los archivos necesarios para cargar y ejecutar el sistema, a continuación se presenta una lista de los directorios de este sistema de archivos:

/dev Contiene los archivos especiales de bloques y caracteres que están generalmente asociados con rutinas de dispositivos de hardware.

/etc Almacena muchas de las órdenes y directorios de administración básica. Dentro de él se pueden encontrar algunos subdirectorios y archivos importantes como:

- **/etc/cron.d** Contiene archivos del comando cron activos.
- **/etc/default** Contiene archivos que asignan ciertos parámetros de sistema por omisión, tales como límites a los intentos del comando su, la longitud de las contraseñas y su envejecimiento.
- **/etc/init.d** Es una localización de almacenamiento para archivos utilizados cuando se cambia de estado de sistema.
- **/etc/lp** Contiene archivos de configuración de impresora local.
- **/etc/mail** Contiene archivos de administración de correo electrónico locales.
- **/etc/rc?.d** Contiene la ubicación actual para los archivos utilizados cuando cambia de estado el sistema; el signo de interrogación (?) se sustituye por cada uno de los estados de sistema válidos.
- **/etc/saf** Contiene archivos de administración de SAF (Service Access Facilities) local.

/export Por default este directorio lo utiliza NFS como raíz del árbol del sistema de archivos exportado.

/lost+found Es usado por la orden fsck para salvar archivos y directorios que están asignados, pero no referenciados en algún punto del sistema de archivos.

/mnt Es donde se deberían montar sistemas de archivos para uso temporal.

/opt Este directorio, si existe, es el punto de montaje desde el que se instalan los paquetes de aplicaciones adicionales.

/sbin Contiene ejecutables usados en el arranque y en la recuperación manual de un fallo del sistema, además podría estar compartido con otras computadoras.

/tmp Contiene archivos temporales.

El sistema de archivos /home

Esta es la ubicación por defecto de los directorios propios de cada usuario. Contiene el directorio de presentación y el árbol de subdirectorios para cada usuario.

El sistema de archivos /usr

El sistema de archivos /usr contiene órdenes y bases de datos de administración del sistema que pueden ser compartidas. En este directorio podemos encontrar los siguientes subdirectorios o archivos importantes:

`/usr/bin` Contiene órdenes públicas y utilidades del sistema.

`/usr/include` Contiene archivos de cabecera públicos para programas C.

`/usr/lib` Contiene bibliotecas públicas, demonios y bases de datos dependientes de la arquitectura utilizados para procesar peticiones en lp, mail, en copias de seguridad y en administración general del sistema.

`/usr/share` Contiene archivos independientes de la arquitectura que pueden ser compartidos. Éste directorio y sus subdirectorios contienen información en archivos de texto plano que pueden ser compartidos entre todas las computadoras que ejecuten el sistema UNIX.

`/usr/sadm/skel` Contiene archivos que son automáticamente instalados en el directorio propio de un usuario cuando el usuario se añade al sistema con la orden `useradd -m`.

`/usr/ucb` Contiene archivos para el paquete de compatibilidad BSD, como archivos de cabecera y bibliotecas.

El sistema de archivos /var

Este sistema de archivos contiene archivos y directorios que sólo pertenecen a la administración de la computadora local y, por tanto, probablemente no serán compartidos con otras computadoras. En general, este sistema de archivos contiene registros de las actividades de la computadora, archivos spool (donde los archivos esperan a ser transferidos o impresos) y archivos temporales. Algunos de los subdirectorios y archivos que se pueden encontrar son:

`/var/adm` Contiene archivos de contabilidad y presentación del sistema.

`/var/cron` Contiene el archivo de registro cron.

`/var/lp` Contiene archivos de registro de la actividad de impresión.

`/var/mail` Contiene el archivo de correo de cada usuario.

- `/var/news` Contiene archivos de noticias del sistema.
- `/var/preserve` Contiene archivos de copia de seguridad para los editores `vi` y `ex`.
- `/var/sadm` Contiene archivos utilizados para servicio de copia de seguridad y de restauración.
- `/var/saf` Contiene archivos de registro para la facilidad de acceso a servicios.
- `/var/spool` Contiene archivos `spool` temporales. Los subdirectorios se utilizan para almacenar las peticiones de `cron`, `lp`, y `mail`.
- `/var/tmp` Contiene archivos temporales.
- `/var/uucp` Contiene archivos de registros y archivos relacionados con la seguridad usados por las utilidades básicas de red.

3.14 Instalación de software

El software de Solaris es distribuido en paquetes. Los paquetes contienen su descripción, una descripción de la relación del sistema destino, archivos que serán instalados y programas de pre y post instalación.

Todo el software del sistema operativo Solaris viene distribuido como paquetes. Durante la instalación, los paquetes son seleccionados y adicionados automáticamente desde el CD-ROM.

Antes de la instalación del sistema operativo, los paquetes de software son adicionados al sistema usando los paquetes de administración o una interfaz gráfica llamada `Admintool`.

El comando `pkginfo`

Es probable que en algún momento necesitemos información de los paquetes que se encuentran instalados en nuestro equipo. El comando `pkginfo` es usado para desplegar información de los paquetes de software que están instalados en el sistema. Por ejemplo:

```
# pkginfo | more
```

application	SUNWaxg	Solaris XGL 3.3 AnswerBook
application	SUNWadm	Solaris 7 System Administrator Collection
System	SUNWab2m	Solaris Documentation Server Lookup
System	SUNWab2r	Solaris Documentation Server
System	SUNWab2s	Solaris Documentation Server
System	SUNWab2u	Solaris Documentation Server
application	SUNWabda	Sun Ultra 5/10 Hardware AnswerBook
application	SUNWabe	Solaris 7 User Collection

Formato del comando

pkginfo -d ruta del dispositivo - **l** pkg_name

Opciones del comando

-d device (dispositivo) Define al dispositivo en el que el software se encuentra. El dispositivo puede ser la ruta de un directorio, cinta, diskette u otro.

-l Usando la opción l se puede obtener toda la información disponible acerca del paquete designado.

pkg_name Designa el nombre del paquete.

Se puede determinar cuántos paquetes existen en el sistema con la siguiente orden:

pkginfo | wc -l

El comando pkgrm

En algunas circunstancias puede ser necesario eliminar uno o varios paquetes del sistema, para realizar esto existe el comando pkgrm.

El comando pkgrm permite borrar los paquetes que se encuentran instalados dentro del sistema, bastará con darle el comando y el nombre del paquete, después de esto el sistema preguntará si se está seguro de querer eliminar dicho paquete.

Formato del comando

pkgrm nombre del paquete

Ejemplo del comando pkgrm

pkgrm SUNWaudio

The following package is currently installed:

```
SUNWaudio      Audio applications
                (sparc) 3.6.4, REV=1.98.08.13
```

Do you want to remove this package? **Y**

Removing installed package instance <SUNWaudio>

Verifying package dependencies.

WARNING:

The <SUNWolrte> package depends on the package currently being removed.

WARNING:

The <SUNWolaud> package depends on the package currently being removed.

WARNING:

The <SUNWoldev> package depends on the package

```
currently being removed.  
WARNING:  
The <SUNWxkey> package depends on the package  
currently being removed.  
Dependency checking failed.  
Do you want to continue with the removal of this package [ y, n, ?, q ]
```

El comando pkgadd

Cuando se necesite adicionar uno o varios paquetes al sistema, se utiliza el comando pkgadd.

Formato del comando

```
pkgadd -d ruta del dispositivo pkg_name
```

Ejemplo del comando pkgadd con CD-ROM.

```
# pkgadd -d /cdrom/cdrom0/s0/Solaris_2.7/Product SUNWaudio
```

```
Processing package instance <SUNWaudio> from  
</cdrom/sol_7_sparc/s0/Solaris_2.7/Product>
```

```
Audio applications  
(sparc) 3. 6. 4, REV=1.98.08.13  
Copyright 1998 Sun Microsystems, Inc. All rights reserved.  
Using </> as the package base directory.  
## Processing package information.  
## Processing system information.  
2 package pathnames are already properly installed.  
## Verifying package dependencies.  
Verifying disk space requirements.  
## Checking for conflicts with packages already installed.  
## Checking for setuid/setgid programs.
```

```
This package contains scripts which will be executed with super-user  
permission during the process of installing this package.
```

```
Do you want to continue with the installation of <SUNWaudio> [ y, n, ? ] y
```

```
## Installing part 1 of 1.
```

```
Installation of <SUNWaudio> was successful.
```

El comando pkgchk

El comando pkgchk verifica que los atributos y contenidos del nombre de la ruta son correctos comparando los valores con los especificados en el archivo del sistema.

Formato del comando

pkgchk -p ruta del archivo

Este comando puede verificar un paquete entero o sólo una ruta particular si la opción **-p** es usada. No muestra salida alguna si el paquete está instalado y no detecta diferencia alguna.

Ejemplo:

```
# pkgchk SUNWaudio
#
# pkgchk -p /etc/group
```

```
ERROR: /etc/group
File size <278> expected <282> actual
File cksum <23586> expected <24038> actual
```

El archivo /etc/group ha sido cambiado de tamaño,

El archivo /var/sadm/install/contents

El archivo contents guardan las rutas donde se encuentran los paquetes de nuestro sistema permitiendo verificar los paquetes de instalación e identificar la localización del directorio de un comando con las siguientes cadenas.

Verificación de un paquete de instalación.

```
# pkginfo SUNWcsu
```

```
system          SUNWcsu          Core Sparc, (U
```

Identificación del directorio de instalación de un comando mediante el comando grep.

```
# grep comando /var/sadm/install/contents
```

```
/usr/bin/showrev f none 0755 root sys 41076 51113 867876437 SUNWadmc
```


El comando `pkgadd` actualiza el archivo `/var/sadm/install/contents` cuando se adiciona un nuevo paquete, el cual contiene las rutas de todos los paquetes instalados en el sistema.

Esto hace posible identificar el paquete o los paquetes que contienen un archivo particular o archivos relacionados a un comando. El comando `pkgrm` usa el archivo `/var/sadm/install/contents` para determinar donde se localizan los archivos y lo actualiza después de ser removidos los paquetes.

Los paquetes spooling

Los paquetes pueden ser copiados desde el CD-ROM sin ser instalados, estos pueden ser almacenados dentro del sistema. Por ejemplo un servidor puede copiar los paquetes del software y almacenarlos para que un sistema sin CD-ROM pueda montar los paquetes del software desde el servidor de instalación.

Se pueden almacenar los paquetes utilizando la opción `-s` y la palabra `spool`, de esta manera los paquetes serán guardados automáticamente en el directorio `/var/spool/pkg`. Ejemplo:

```
#pkgadd -d /cdrom/cdrom0/s0/Solaris_2.7/Product -s spool SUNWaudio
```

```
Transferring <SUNWaudio> package instance
```

Es posible especificar el directorio donde se quiera almacenar los paquetes indicandolo con la opción `-s` y la ruta del directorio. Ejemplo

```
# pkgadd -d /cdrom/cdrom0/s0/Solaris_2.7/Product -s /export/pkg SUNWaudio
```

```
Transferring <SUNWaudio> package instance
```

También es posible borrar los paquetes almacenados del directorio `spool` con la opción `-s`. Ejemplo

```
# pkgrm -s /export/pkg SUNWaudio
```

3.15 Instalación de parches

Un parche es un conjunto de archivos o directorios que reemplazan o actualizan archivos o directorios para la ejecución de algún programa de software.

Un parche se encarga de hacer actualizaciones al entorno del sistema para que cada una de las funciones pueda reconocer las nuevas aplicaciones o funciones.

A los parches se les asigna un número y son empaquetados en un directorio llamado con el número del parche. Por ejemplo, si el número del parche es 101945 y la versión es 34 el directorio será llamado 101945-34.

El directorio donde se puede encontrar información acerca de los parches instalados dentro del sistema esta en /var/sadm/patch. Por ejemplo:

```
# ls /var/sadm/patch
103558-05      103594-04      103630-01      103663-01      103683-01
103696-01      103817-01      103582-01      103612-06      103640-03
```

Es recomendable que cuando se vaya a instalar un nuevo parche primero sea descargado en el archivo tmp y de ahí se haga la instalación. En la versión de sistema operativo 2.6 y anteriores se utiliza el comando patchadd para instalarlo, por ejemplo:

```
# cd /tmp
# ls
105160-01      ps_data      sdt_f18zKOL_
```

```
patchadd 105161-01
Checking installed packages and patches...
Verifying sufficient filesystem capacity (dry run method)
Installing patch packages...
```

```
Patch number 105161-01 has been successfully installed.
See /var/sadm/patch/105161-01/log for details
```

```
Patch packages installed:
SUNWdtbas
```

En las versiones posteriores a la 2.6 se utiliza el comando installpatch, por ejemplo:

```
# cd /tmp/105050-01
# ls
Install.info      SUNWcsu      backoutpatch
README.105050-01  SUNWscpu     installpatch
```

```
./installpatch
Checking installed packages and patches...
Generating list of files to be patched...
Verifying sufficient filesystem capacity (exhaustive method)
Installing patch packages...
```

```
Patch number 105050-01 has been successfully installed.
See /var/sadm/patch/105050-01/log for details
```

```
Patch packages installed:
SUNWcsu
SUNWscpu
```

TESIS CON
FALLA DE ORIGEN

Cuando se lista el contenido de un parche se presenta información con relación a él, por ejemplo:

```
# cd 104641-01
```

```
# ls
Install.info      SUNWolrte      backoutpatch
README.104641-05 SUNWolslb      installpatch
```

Al listar el contenido del parche aparecen los archivos `Install.info` y `README`, estos archivos listan información detallada acerca de los requisitos para la instalación y variantes sobre los procedimientos de instalación.

Los archivos `SUNWolrte` y `SUNWolslb` indican los paquetes de software que son modificados por este parche. Estos son los directorios que contienen los nuevos archivos a reemplazar.

`Installpatch` y `backoutpatch` son archivos que se utilizan para la instalación y de ser necesario se regresa al parche original.

En algún momento se puede necesitar eliminar un paquete de parches, para ello se puede utilizar el comando `patchrm` por ejemplo:

```
# patchrm 104945-02
```

```
Checking installed packages and patches...
```

```
Patch 104945-02 has been backed out.
```

Después de determinar que puede ser necesario la instalación de algún parche, se puede verificar que éste no exista en el sistema con el comando `showrev` o el comando `patchadd`, para los dos se utiliza la opción `-p` por ejemplo:

```
# patchadd -p
```

```
Patch: 105160-05 Obsoletes: Requiere: Incompatibles:
Packages: SUNWdtbas
```

```
# showrev -p
```

```
Patch: 105160-05 Obsoletes: Requiere: Incompatibles:
Packages: SUNWdtbas
```

Existe una dirección en Internet donde se puede obtener información y adquirir parches para adicionar a nuestro sistema, la dirección es : <http://sunsolve.sun.com>

3.16 Procesos del sistema

Un proceso es un programa en ejecución. Sin embargo, un programa puede crear nuevos procesos, para un programa dado puede haber uno o varios procesos en ejecución. El sistema puede identificar los procesos que se corren en él por medio de un número que le asigna a cada proceso (PID).

Al nivel más bajo, un proceso se crea mediante una llamada del sistema `fork`. (una llamada de sistema es una subrutina que hace que el núcleo proporcione algún servicio a un programa). `fork` crea un proceso en ejecución separado, pero casi idéntico. El proceso que hace la llamada de sistema `fork` se denomina proceso padre, el proceso que se crea después de la llamada se denomina proceso hijo. Los dos procesos tienen el mismo entorno, las mismas especificaciones de manejo de señales, los mismos identificadores de grupo y de usuario y la misma clase y prioridad, pero diferentes números de proceso.

“Existe una metáfora especial que se aplica a los procesos en el sistema. Los procesos tienen *vida*: están *vivos o muertos*; son *generados (nacidos)*, *mueren*; pueden convertirse en *zombies* o convertirse en *huérfanos*. Son *padres o hijos*, y cuando deseamos suprimirlos, los *matamos*.”¹⁴

El comando `ps`

Para ver los procesos que están corriendo dentro del sistema se usa la orden `ps` (estado de proceso). La orden `ps` lista todos los procesos activos en ejecución en la máquina. Si se usa `ps` sin opciones, se obtiene información referente a los procesos asociados con la terminal. Por ejemplo, la salida de `ps` muestra el ID de procesos (PID), el ID de la terminal (TTY), la cantidad de tiempo de CPU en minutos y segundos que la orden ha consumido y el nombre de la orden. Por ejemplo:

`ps`

PID	TTY	TIME	COMD
3211	term/41	0:05	ksh
12326	term/41	0:01	ps
12233	term/41	0:20	ksh
9046	term/41	0:02	vi

Este usuario tiene cuatro procesos asociados al ID de terminal `term/41`; hay dos shells Korn, `ksh`, una sesión de edición `vi` y la propia orden `ps`.

“Los números de ID de procesos se asignan secuencialmente cuando se crean los procesos. El proceso 0 es un proceso del sistema que se crea al principio cuando se arranca el sistema y el proceso 1 es el proceso `init` a partir del cual se generan todos los restantes. Los otros IDs de proceso comienzan en 2 y prosiguen numerando cada nuevo proceso con el siguiente número disponible. Cuando se alcanza el máximo número de ID de proceso, la numeración comienza de nuevo saltándose cualquier número que todavía esté en uso. El máximo número puede variar, pero generalmente se fija en 32767.”¹⁵

¹⁴ Ídem. p. 752

¹⁵ Ídem. p. 753

Opciones del comando ps

Cuando se utiliza el comando ps con opciones, es posible controlar la información visualizada referente a los procesos en ejecución.

f La opción `-f` proporciona un listado completo de los procesos: La primera columna identifica el nombre de presentación (UID) del usuario, la segunda columna (PID) es el ID del proceso y la tercera (PPID) es el ID del proceso padre —el número de ID del proceso que generó a éste—. La columna C representa un índice de utilización reciente del procesador, que es utilizado por el núcleo para planificación. STIME es el tiempo de inicio del proceso en horas, minutos y segundos; un proceso con más de 24 horas de antigüedad se da en meses y días. TTY es el número de ID de la terminal. TIME es el tiempo acumulado de el CPU consumido por el proceso y COMMAND es el nombre de la orden que está siendo ejecutada. Por ejemplo:

ps -f

UID	PID	PPID	C	STIME	TTY	TIME	COMMAND
root	17118	3211	0	15:57:02	term/41	0:01	/usr/bin/vi
root	3211	1	0	15:16:34	term/41	0:00	/usr/bin/ksh
root	2187	17118	0	16:56:09	term/41	0:00	-ksh
root	4764	2187	0	16:59:00	term/41	0:00	ps -f

u Se pueden conocer los procesos que está corriendo algún usuario con la opción `-u` usuario por ejemplo:

ps -f -u ana

UID	PID	PPID	C	STIME	TTY	TIME	COMMAND
ana	17118	3211	0	15:57:02	term/41	0:01	ksh
ana	3211	1	0	15:16:34	term/41	0:00	vi resumen

l La opción `-l` proporciona una forma larga del listado de ps. El listado largo repite la información de la opción f y adiciona algunas columnas, la primera columna (F), especifica un conjunto de indicadores hexadecimales aditivos que identifican características del proceso; por ejemplo, el proceso ha terminado, 00; el proceso es un proceso del sistema, 01; el proceso está en memoria principal, 08; el proceso está bloqueado, 10. La segunda columna identifica el estado actual del proceso.

Estado del proceso (S)

Abreviatura	Significado
O	Proceso ejecutándose
S	Proceso durmiendo
R	Proceso ejecutable en cola
I	Proceso inactivo, en creación
Z	Zombie
T	Proceso detenido y siendo trazado
X	Proceso a la espera de más memoria

La columna PRI contiene el valor de prioridad del proceso (un valor superior significa una prioridad inferior) y NI es el valor nice del proceso. ADDR representa la dirección inicial en memoria del proceso. SZ es el tamaño, en páginas, del proceso en la memoria. Ejemplo:

ps -l

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	COMD
100	S	4392	17118	3211	0	30	20	1c40368	149	10d924	term/41	0:01	vi
0	O	4392	4847	2178	37	3	20	32686d0	37		term/41	0:01	ps
100	S	4392	3211	1	0	30	20	2129000	52	1161a4	term/41	0:01	ksh
100	S	4392	21788	17118	8	30	20	35a7000	47	1176a4	term/41	0:01	ksh

e Si se utiliza la opción -e, se visualizan todos los procesos que están ejecutándose en el sistema. La orden ps -e muestra el ID de proceso, la terminal (aparece una ? si el proceso no está asignado a ninguna terminal), la hora y el nombre de la orden. Ejemplo:

ps -e

PID	TTY	TIME	COMMAND
0	?	0:34	sched
1	?	41:55	init
2	?	0:00	vhand
3	?	20:56	bdflush
23724	console	0:03	sh

El comando kill

En algún momento puede quererse detener un proceso mientras está en ejecución. Por ejemplo, puede decidirse no completar un proceso que se ha iniciado, porque esté haciendo algo no deseado. Si un proceso está activamente ejecutándose, sólo se han de presionar las teclas control y d ó c simultáneamente. Sin embargo, no se puede terminar de este modo un proceso subordinado o uno asociado a una terminal diferente.

Para terminar ese proceso, o "matarlo", se usa el comando kill, dando el ID de proceso como argumento. Por ejemplo, para eliminar el proceso con PID 2312 debe teclearse

kill 2312

Esta orden envía una señal al proceso. En particular cuando se utiliza sin argumentos, el comando kill envía una señal 15 al proceso. La señal 15 es una señal de terminación de software que se emplea para detener procesos.

Existen algunos procesos, que no mueren cuando reciben esta señal. Se pueden eliminar los procesos que ignoran la señal 15 con la señal -9, de este modo se envía al proceso la señal 9, que es la eliminación incondicional.

Procesos padre e hijos

Cuando se escribe una línea de órdenes en el sistema, el shell maneja su ejecución. Si la orden es una orden interna conocida por el shell como `exit` o `echo`, se ejecuta internamente sin crear un nuevo proceso. Si la orden no es interna, el shell la trata como un archivo ejecutable. El shell actual usa la llamada de sistema `fork` y crea un proceso hijo que ejecuta la orden. El proceso padre, espera hasta que el hijo muere y luego vuelve para leer la siguiente orden.

Normalmente, cuando el shell crea el proceso hijo, ejecuta una llamada de sistema `wait`. Esto suspende la operación del shell padre hasta que recibe una señal indicando la muerte de un hijo. En ese momento el proceso padre despierta y busca una nueva orden.

Cuando se escribe una orden que tarda mucho tiempo en ejecutarse, generalmente es necesario esperar hasta que termine. Es posible ejecutar múltiples procesos al mismo tiempo poniendo trabajos en modo subordinado colocando un signo de `&` al final de la cadena de instrucciones, de éste modo se le indica al shell que ejecute la orden en modo subordinado.

Cuando el shell ve el `&` al final de la orden, genera un proceso hijo para ejecutar la orden, pero no ejecuta la llamada de sistema `wait`. En vez de suspender la operación hasta que el hijo muera, el shell padre reanuda el procesamiento de órdenes inmediatamente.

Demonios

Los demonios son procesos que no están conectados a un terminal, pueden ejecutarse en un modo subordinado y realizar trabajos útiles. Normalmente pueden encontrarse varios demonios en los sistemas: los demonios de usuario, que son fabricados por él mismo y los demonios de sistema que gestionan la planificación y administración.

Se dice que los demonios son procesos que están en constante movimiento dentro del sistema, en espera de que se les asigne una tarea determinada. Cada uno de estos demonios está controlado por `cron`, que es asimismo ejecutado por `init`.

Procesos zombie

Normalmente, los procesos del sistema terminan utilizando la llamada de sistema `exit`. La llamada `exit` devuelve el valor de estado al proceso padre. Cuando un proceso emite la llamada `exit`, el núcleo desactiva todo el manejo de señales asociado con el proceso, cierra todos los archivos, libera todos los recursos, libera la memoria utilizada, asigna los posibles hijos del proceso que acaba, envía la señal muerte de un hijo al proceso padre y convierte al proceso al estado zombie. Un proceso en estado zombie no está vivo, no utiliza ningún recurso ni efectúa ningún trabajo, pero no se le permite que muera hasta que la salida sea reconocida por el proceso padre.

Si el padre no reconoce la muerte del hijo (porque el padre éste ignorando la señal o porque el propio padre esté colgado), el hijo permanece como proceso zombie. Estos procesos aparecen en el listado de ps -f con <defunct> en el lugar de la orden.

Debido a que un proceso zombie no consume tiempo de el CPU y no está ligado a ninguna terminal, los campos STIME y TTY están en blanco.

En algunos sistemas es necesario liberar a los procesos zombies pero en esta versión son liberados automáticamente.

El comando at

Como sistema de tiempo compartido, el núcleo del sistema controla directamente cómo se planifican los procesos. Existen órdenes a nivel de usuario que permiten especificar cuándo ejecutar los procesos.

Utilizando el comando at puede especificarse cuándo deberían ejecutarse las órdenes. at lee órdenes de la entrada estándar y las planifica para ejecución. Normalmente, la salida y el error estándares se envían por correo, a menos que sean redirigidos a algún otro lugar.

Formato del comando

```
at [-m] [-r job] time date
```

Opciones

-m	Manda un mensaje al terminar el comando at.
-r	Elimina un proceso creado con el commando at.
time	Especifica el tiempo.
date	Especifica el mes seguido por el día.

El comando at puede utilizarse para ordenarle al sistema que ejecute una orden en un tiempo especificado, por ejemplo. Si se desea que a las 8:45 pm busque los archivos core del directorio /export/home y que sean borrados se deberá ejecutar la siguiente orden:

```
# at 8:45 pm
at> find /export/home -name core -exec rm {} \;
at> (control + D)
```

```
commands will be executed using /bin/ksh
job 891550468.a at Thu Apr 2 13:54:28 2000
```


Existe en el sistema el archivo `/etc/cron.d/at.deny` que contiene quiénes son los usuarios que no pueden utilizar el comando `at`.

Si se desean conocer los procesos `at` que están en espera de ser ejecutados se utiliza el comando `atq`. Por ejemplo:

```
# atq
```

Rank	Execution	Date	Owner	Job	Queue	JobName
1 st	Apr 2, 2000	13:54	root	891550468.a	a	stdm

Si en algún momento se deseara eliminar el proceso se utiliza la opción del comando `at -r` y el número de job dado por el sistema al momento de la creación. Por ejemplo:

```
# at -r 891550468.a
```

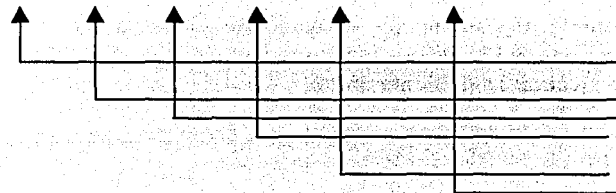
El comando crontab

Cron es un demonio de sistema que ejecuta órdenes en instantes específicos. La información de órdenes y de planificación están mantenidas en el directorio `/var/spool/cron/crontabs`. Cada usuario que tiene permisos para planificar órdenes con cron tiene un archivo `crontab`. El demonio despierta periódicamente para ejecutar los trabajos que estén planificados para ese momento.

El archivo `/etc/cron.d/cron.allow` contiene la lista de todos los usuarios que tienen permitido realizar entradas en el `crontab`, pero este archivo no existe por default en el sistema, requiere de su creación por parte del administrador. Si el archivo no existe puede estar el archivo `/etc/cron.d/cron.deny` que contiene los usuarios que tienen prohibido correr el `crontab`. Si ninguno de los dos archivos existe sólo el superusuario podrá ejecutar el `crontab`.

Las entradas a un archivo `crontab` tienen seis campos. El primer campo es el minuto en que la orden va a ejecutarse, el segundo campo es la hora, el tercero el día del mes, el cuarto el mes del año, el quinto el día de la semana, y el sexto la cadena de orden a ejecutar. Los asteriscos actúan como comodines. Por ejemplo:

```
10 3 * * 0 /usr/lib/newsyslog
```



- El minuto, puede tener el valor entre 0 y 59
- La hora, puede tener el valor entre 0 y 23
- El día del mes, puede tener el valor entre 1 y 31
- El mes, puede tener el valor entre 1 y 12
- El día de la semana, puede tener el valor entre 0 y 6
- El comando, contiene el comando a ser ejecutado.

El comando pgrep y pkill

El comando pgrep y pkill remplazan la combinación de los comandos ps, grep, egrep, awk y kill que eran usados en versiones anteriores del sistema Solaris 7.

El comando pgrep examina la actividad de los procesos en el sistema y despliega los IDs de los procesos que se ajustan a los atributos especificados en la línea de comandos.

El comando pkill trabaja exactamente igual que el comando pgrep, sólo que éste termina los procesos, generalmente usando la señal -15.

Formato del comando

```
pgrep [-opción] pattern  
pkill [-opción] pattern
```

Opciones de los comandos

- U Muestra el número identificador del usuario.
- t Muestra la terminal donde el proceso está corriendo.
- l Despliega el nombre del proceso.
- G Muestra el nombre identificador del grupo.
- n Muestra el proceso más reciente.

Ejemplo del uso del comando pgrep y pkill

```
# pgrep -l mail  
399 sendmail  
942 dtmail
```

```
# pkill dtmail
```

La primera línea localiza a todos los procesos de mail que están corriendo, despliega el número identificador y el nombre del proceso. La segunda línea indica el proceso que será terminado por medio del nombre.

Se puede usar la opción -U para prevenir que accidentalmente se pueda matar un proceso perteneciente a otro usuario aún cuando el comando se esté ejecutando por el superusuario.¹⁶

```
# pkill -U bevw dtmail
```

¹⁶ Cfr. Sun Educacional Services. *Solaris 7 System Administration I*. Cap. 12

El comando nohup

Cuando una terminal es desconectada, el núcleo manda una señal a todos los procesos que estaban asociados a la terminal, siempre que el shell no tenga control de los trabajos. La intención de esta señal es hacer que todos los procesos restantes terminen. Sin embargo, con frecuencia se desea que una orden continúe su ejecución después de cerrar la sesión. Por ejemplo, puede desearse continuar un proceso que está formateando un archivo sin tener que permanecer conectado.

Para asegurarse que un proceso permanezca vivo después de que se despide, se utiliza el comando nohup.

Formato del comando

```
# nohup [orden]
```

El comando nohup se refiere únicamente a la orden que le sigue inmediatamente. Si se emite una cadena como ésta:

```
# nohup date; who; ps -ef
```

Sólo la primera orden ignorará la señal de colgado, las órdenes restantes de la línea acabarán después de colgar. Para utilizar el comando con múltiples órdenes, se puede preceder cada una de ellas con el comando o se colocan todas las órdenes en un archivo y se usa nohup para proteger el shell que está ejecutando las órdenes.

```
# cat archivo  
date  
who  
ps -ef  
# nohup sh archivo
```

3.17 Adición de usuarios

Una de las ventajas que proporcionan los equipos Sun y el sistema operativo Solaris es que se pueden tener a varios usuarios dentro del equipo realizando trabajos específicos.

Se puede utilizar el comando useradd para identificar a un nuevo usuario en la computadora y permitir que él acceda a la misma. Esta orden evita el tener que editar manualmente los archivos /etc/passwd y /etc/shadow. El comando useradd puede ser utilizado con varias opciones para indicarle características específicas que deberá tener el nuevo usuario que deseamos adicionar al sistema.

Opciones del comando useradd

-u (uid) Esta opción define el ID del nuevo usuario. El uid es por default el siguiente número disponible por encima del número más alto actualmente asignado en el sistema.

-o La opción **-o** se utiliza junto con la opción **-u** para indicarle al sistema que se desea asignar a un nuevo usuario con un uid que no sea único. Se puede querer hacer esto si se desea que varios usuarios diferentes tengan los mismos permisos de acceso a archivos, pero diferentes nombres y directorios propios.

-g (grupo) Esta opción define un número de ID de grupo o nombre de grupo ya existente. Los valores por default que entrega el sistema son 1 para el ID de grupo y other para el nombre de grupo.

-d (dir) Esto define el directorio propio del nuevo usuario. El valor por default, cuando se adiciona el usuario es /home/nombredeusuario.

-s (shell) Establece el nombre de camino completo del shell de presentación del usuario. El shell por default, cuando se adiciona el usuario es /bin/sh.

-c (comentario) Se utiliza para establecer cualquier comentario que se desee añadir a la entrada del usuario en el archivo /etc/passwd.

-k (skel_dir) Define el directorio que contiene información esquemática (como el .profile) a copiar en el directorio propio del nuevo usuario. El directorio esquemático por default, cuando se adiciona un nuevo usuario en el sistema es el /etc/skel.

-e (expira) Define la fecha en que una presentación expira. Esta opción es útil para crear accesos temporales, la expiración por default, cuando se adiciona un usuario es 0 (no expira).

-f (inactiva) Establece el número de días que una presentación puede estar inactiva antes de que sea declarada inválida. El valor por default cuando se adiciona un usuario es 0 (no es inválida).

Ejemplo de adición de un nuevo usuario con el comando useradd

```
# useradd -uo 0 -s /bin/ksh -c "soporte técnico" -d / SOP
```

El ejemplo anterior permite adicionar un nuevo usuario que tiene los mismo permisos del superusuario, tiene el shell ksh, como comentario tiene soporte técnico, se encuentra su directorio de trabajo en raíz y se llama sop.

Una vez que se ha adicionado un nuevo usuario dentro del sistema es necesario notificarle para que éste le asigne una clave de acceso que sólo él conozca para prevenir que otro usuario pueda leer sus archivos.

Si es necesario borrar algún usuario existente dentro del sistema, se utiliza el comando `userdel`, por ejemplo si se desea borrar el usuario `sop` se deberá hacer de la siguiente manera:

```
#userdel sop
```

Adición de un grupo

La creación de grupos de trabajo puede ser útil cuando se desea que varios usuarios tengan permisos para un conjunto particular de archivos o directorios.

Por ejemplo, puede quererse asignar los usuarios que están escribiendo manuales a un grupo llamado `textos` y darles permisos para un directorio que contenga documentos, para crear un grupo se utiliza el comando `groupadd` seguido del nombre que se desea, por ejemplo:

```
#groupadd textos
```

La cadena anterior añadirá el nombre `textos` al archivo `/etc/group` y el sistema le asignará un número de ID de grupo.

Una vez creado un grupo, pueden asignarse usuarios a éste. Para asignar un nuevo usuario a un grupo existente, se utilizará el comando `useradd` mencionado anteriormente, por ejemplo:

```
#useradd -g textos abc
```

El ejemplo anterior añadirá un nuevo usuario llamado `abc` al grupo de trabajo `textos`.

Si se decide que no se necesita más un grupo que se había añadido previamente, puede eliminarse con el comando `groupdel` y el nombre del grupo, por ejemplo:

```
#groupdel textos
```

Variables de shell

El shell dispone de un mecanismo para definir variables que se pueden utilizar para contener información usada por los programas del sistema o para uso propio. Las variables de shell se pueden utilizar para personalizar o particularizar la información relativa a nombres de directorios y de archivos que necesitan los programas y para particularizar la forma en que los programas (incluyendo el propio shell) interactúa con el usuario.

La siguiente es una lista de algunas variables de shell más comunes, incluyendo aquellas que son fijadas automáticamente al sistema, son las variables denominadas de entorno ya que configuran distintos aspectos del entorno del usuario.

HOME Contiene el nombre de camino absoluto de su directorio de trabajo. Está definida automáticamente y fijada al directorio de trabajo como parte del proceso de inicio de sesión. El propio shell utiliza esta información para determinar el directorio al que cambiar cuando se teclea la orden `cd` sin argumento.

PATH Lista por orden los directorios en los que el shell busca para encontrar el programa a ejecutar cuando se teclea una instrucción. Contiene una lista de nombres de directorios separados por dos puntos. Un **PATH**, por omisión, es definido por el sistema, pero la mayor parte de los usuarios lo modifican para añadir directorios de ordenes adicionales. Un campo vacío en la cadena **PATH** significa buscar en el directorio actual.

Un ejemplo típico de **PATH** particularizado es el siguiente:

```
PATH=/usr/bin:/usr/ucb:/etc:/usr/sbin:/usr/openwin/bin:  
export PATH
```

El ejemplo anterior significa que cuando se introduzca una orden, el shell buscará en primer lugar el programa en el directorio `bin`, después en el subdirectorio `ucb` y así sucesivamente, finalmente en el directorio actual indicado por el campo vacío al final.

CDPATH Es similar a **PATH**. Lista en orden los directorios en los que busca el shell para encontrar un subdirectorio a cambiar cuando se utiliza la orden `cd`. Los directorios que busca el shell se listan de la misma forma que los directorios de la variable **PATH**.

PS1 y **PS2** Definen los signos de sus peticiones de orden primaria y secundaria respectivamente. Sus valores por omisión son `$` para **PS1** y `>` para **PS2**.

LOGNAME Contiene el nombre del usuario. Es fijado automáticamente por el sistema.

MAIL Contiene el nombre del directorio en el que se coloca el nuevo correo. El shell utiliza esta variable para notificar cuándo se añade una nueva información al directorio.

MAILFILE Utiliza la orden `mailx` para encontrar dónde colocar el nuevo correo.

SHELL Contiene el nombre del programa de shell. Es utilizado por el editor `vi` y por otras ordenes interactivas para determinar qué programa de shell ejecutar cuando se emite una orden de escape de shell. (Un escape de shell es una orden que se introduce mientras se trabaja en un programa interactivo como `vi` o `mailx` que temporalmente interrumpen el programa y vuelve al shell). Define el nombre de camino de shell `/bin/ksh`.

TERM Es utilizada por el editor `vi` y otros programas orientados a pantalla para obtener información sobre el tipo de terminal que se está utilizando. Esta información es necesaria para permitir a los programas que comparen su salida con las capacidades de la terminal y para interpretar su entrada correctamente. Define el tipo de terminal `vt100`.

El comando mv y cp

Para mantener organizado el sistema de archivos se necesita desplazar y renombrar archivos. Por ejemplo, se puede utilizar un directorio para borradores de documentos y moverlos desde éste hasta un directorio final cuando estén acabados.

Se puede renombrar un archivo para darle un nombre que resulte más informativo o más fácil de recordar, o que refleja los cambios en su contenido o estado. Un archivo se mueve desde un directorio hasta otro con mv (de move). Por ejemplo, la siguiente orden mueve el archivo *nombres* desde el directorio actual hasta el directorio */export/home/dir*:

```
# mv nombres /export/home/dir
```

Se pueden mover varios archivos simultáneamente hacia el mismo directorio destino nombrando en primer lugar todos los archivos a mover y dando al final el nombre del destino. Por ejemplo, la siguiente orden mueve tres archivos al subdirectorio denominado *Capitulo1*:

```
# mv tema1 tema2 tema3 Capitulo1
```

No existe una orden independiente en el sistema operativo para renombrar un archivo. El renombrado de un archivo es parte de su movimiento. Se puede renombrar un archivo cuando se le mueve a un nuevo directorio incluyendo el nuevo nombre de archivo como parte del destino. Por ejemplo, la siguiente orden coloca el archivo *notas* en el directorio *Capitulo1* y le da el nombre *tema4*.

```
# mv notas Capitulo1/tema4
```

Para renombrar un archivo en el directorio actual también se utiliza *mv*, pero con el nombre de archivo nuevo como destino. Por ejemplo, la siguiente cadena renombra *repaso* a *resumen*.

```
# mv repaso resumen
```

Cuando se utiliza mv, se nombra en primer lugar el archivo a mover, después el destino. El destino puede ser un directorio y un nombre de archivo, un nombre de directorio solo, o un nombre de archivo solo. El nombre de un directorio destino puede ser un nombre de camino completo, o un nombre relativo al directorio actual. Si el destino no es un nombre de directorio, mv lo utiliza como nuevo nombre para el archivo del directorio actual.

El movimiento de archivos es muy rápido en el sistema. El contenido actual de un archivo no se mueve, sólo una entrada en una tabla que dice al sistema en qué directorio se encuentran los datos. Por tanto, el tamaño del archivo no influye en el tiempo consumido por la orden mv.

Cuando se utiliza mv se debe de tener cuidado de no cometer una serie de errores comunes. Por ejemplo, cuando se mueve un archivo a un nuevo directorio es bueno comprobar primero que no existe ya un archivo con ese nombre. Si existe, mv sobrescribirá en él el nuevo archivo. Si se comete un error cuando se especifica un directorio destino, puede terminar renombrando el archivo en el directorio actual.

Para ayudar a prevenir la sobreescritura accidental de archivos. La opción `-i` (interactive) hace que `mv` le informe cuando en un movimiento existe la posibilidad de sobrecribir un archivo existente. Se visualiza el nombre del archivo seguido por una interrogación. Si quiere confirmar el movimiento, se pulsará `y`. Cualquier otra entrada aborta el movimiento.

Otra característica es la posibilidad de utilizar `mv` para mover directorios. Una única orden se puede utilizar para mover un directorio y todos sus archivos y subdirectorios, de la misma forma que se mueve un único archivo.

El comando `cp`

Uno de los principales motivos para copiar un archivo es la seguridad, de manera que se pueda modificar un archivo sin preocuparse de la pérdida del original. La orden `cp` es similar a `mv`, excepto que copia archivos en lugar de moverlos o renombrarlos. `cp` sigue el mismo modelo que `mv`: se nombran los archivos a copiar en primer lugar, después el destino. El destino puede ser un directorio, un directorio y archivo, o un archivo en el directorio actual. La orden siguiente hace una copia de seguridad de `doc` y nombra la copia como `doc.bk`.

```
# cp doc doc.bk
```

Después de utilizar la orden `cp` existen dos copias separadas del archivo en el mismo directorio. El original queda inalterado. El contenido del archivo copiado es idéntico al original.

```
# cp doc copia
```

En el ejemplo anterior se debe tomar en cuenta que si el directorio destino ya contiene un archivo denominado `doc.bk`, la copia lo sobrecribirá. Si se invoca al comando `cp` con la opción `-i` se recibirá un aviso antes de sobrecribir un archivo existente. Por ejemplo, si existe un archivo `datos2` en el directorio actual, `cp` avisará que lo sobreescribiera y preguntará si desea continuar.

Para crear una copia de un archivo con el mismo nombre que el original en un nuevo directorio, se utiliza el nombre del directorio como destino, por ejemplo:

```
# cp doc dest
```

Otra opción del comando `cp` es `-r`, que permite copiar una estructura de directorio completa en otro directorio. Si `tema` es un directorio, la siguiente orden copia todos los archivos y subdirectorios de `tema` sobre el nuevo directorio.

```
# cp -r tema nuevo.
```

Cuando se copia un archivo, crea otro con el mismo contenido que el original. Cada copia ocupa espacio adicional en el sistema de archivos y cada una se puede modificar independientemente de las otras.

El comando vi

La mayoría de los usuarios de computadoras gastan más tiempo en la creación y modificación de textos que en cualquier otra cosa. Escribir memorándums, cartas, libros y programas, y crear archivos de texto de cualquier tipo implica un gran esfuerzo. "El sistema Operativo Solaris separa la forma de realizar este trabajo en dos actividades. La creación y modificación de texto lo realizan los *editores*, mientras que el formateo de texto para la visualización y la presentación final lo realizan los *formateadores*. La razón de esta separación es la habitual de UNIX de tener programas flexibles que se centran en hacer bien una cosa".¹⁷

Cuando se está creando texto, es recomendable centrarse en lo que se quiere escribir en un archivo de la computadora. Una vez que se tiene algún material, entonces es útil concentrarse en cómo formatearlo o cómo se quiere que aparezca. El sistema operativo proporciona un conjunto de herramientas útiles para la creación y modificación de texto. En este capítulo se presentará el editor *vi* junto con sus herramientas más utilizadas para la creación de un archivo de texto.

vi es un editor con dos modos separados de *entrada* y de *orden*. Es decir, en el modo de entrada todo lo que se escriba en el teclado se interpreta como entrada a colocar en el archivo que se está editando; en el modo de orden, cualquier cosa que se introduzca mediante teclado se toma como una orden para el editor para permitir moverse por los archivos o cambiar parte de ellos.

Cuando el editor está en modo de entrada, los caracteres que se teclean se introducen como texto en el buffer. Cuando el editor está en modo orden, los caracteres que se teclean son órdenes que controlan la pantalla o cambian el contenido del buffer.

Cuando se está en modo de *entrada*, ESC (escape) le cambia al modo de orden; cualquier cosa que se teclee se introduce en su documento. Si se está en modo de *orden*, varias órdenes (a, A, i, I, o y O descritas más adelante) devolverán al modo de entrada.

Debido a las diferentes características de las terminales, lo primero que debe hacerse antes de utilizar *vi* es especificar el tipo de terminal configurando una variable de entorno. Por ejemplo, el vt100 es un modelo. Si utiliza esa terminal, tecleando la siguiente instrucción:

```
# TERM=vt100
#export TERM
```

Se fija la variable de terminal vt100, quedando disponible esta información a los programas que lo necesiten. Para no tener que teclear la información del terminal cada vez que se abre una sesión, se puede incluir en el archivo *.profile* para fijar automáticamente la terminal a vt100 siempre que se introduzca en el sistema.

¹⁷ Rosen, et al. *UNIX. Sistema V, Versión 4*. p. 197

Para crear un nuevo archivo con el comando vi se realiza el siguiente paso:

vi *nombre del archivo*.

Para insertar texto dentro de un documento pueden utilizarse las siguientes opciones:

Comando	Actividad
a	Inserta texto después del cursor.
A	Inserta texto al final de la línea.
i	Inserta texto antes del cursor.
I	Inserta texto al inicio de la línea.
o	Inserta una nueva línea abajo del cursor.
O	Inserta una línea arriba del cursor.

Los siguientes comando de vi pueden ser utilizados para moverse dentro del documento y poder realizar algún cambio requerido.

Comando	Actividad
←, h ó Back Space	Se mueve el cursor un carácter a la izquierda.
↓ ó j	Se mueve el cursor una línea hacia abajo.
↑ ó k	Se mueve el cursor una línea hacia arriba.
→, l ó Space bar	Se mueve el cursor un carácter hacia la derecha.
w	Mueve el cursor una palabra hacia delante (incluyendo puntuación).
W	Mueve el cursor una palabra hacia delante.
b	Mueve el cursor una palabra hacia atrás (incluyendo puntuación).
B	Mueve el cursor una palabra hacia atrás.
e	Mueve el cursor al final de la palabra donde está colocado.
\$	Mueve el cursor al final de la línea.
0	Mueve el cursor al inicio de la línea.
H	Mueve el cursor al principio de la pantalla.
M	Mueve el cursor a la mitad de la pantalla.
L	Mueve el cursor al final de la pantalla.

Hasta el momento, se han visto los comandos de vi para introducir texto y mover el cursor dentro del mismo, a continuación se verá cómo borrarlo.

Comando	Actividad
x	Borra el carácter donde se encuentra colocado el cursor.
X	Borra el carácter a la izquierda del cursor.
dw	Borra una palabra.
3dw	Borra tres palabras.
dd	Borra toda la línea donde está colocado el cursor.
3dd	Borra tres líneas debajo de donde está el cursor.
D	Borra la línea a partir de donde está colocado el cursor.
dG	Borra al final del archivo a partir de donde está el cursor.
DIG	Borra al inicio del archivo a partir de donde está el cursor.

Para copiar y pegar texto se pueden utilizar los siguientes comandos.

Comando	Actividad
YY	Copia una línea.
P	Pega la línea copiada arriba del cursor.
P	Pega la línea copiada abajo del cursor.

Para salvar los cambios a un archivo o salirse de él sin las modificaciones puede hacerse lo siguiente:

Comando	Actividad
:w	Salva los cambios que están escritos en el buffer.
:w <i>nuevo_archivo</i>	Escribe el buffer a un nuevo archivo.
:wq	Salva los cambios y sale del editor vi.
ZZ	Salva los cambios y sale del editor vi.
:q!	Salva los cambios y sale del editor vi (el signo ! obliga a sobrescribir cuando se es el dueño del archivo y sólo tiene permisos de lectura).

Otras opciones del editor vi que puedes ser de gran ayuda se presentan a continuación:

Comando	Actividad
cw	Cambia una palabra o parte de ella a partir de donde está el cursor y hasta el final de la palabra.
3cw	Cambia tres palabras.
R	Cambia o reemplaza caracteres en la línea.
C	Cambia desde donde está el cursor hasta el final de la línea.
~	Cambia una letra por mayúscula o minúscula según sea el caso.
U	Cancela todos los cambios hechos en la línea.
u	Cancela el último cambio hecho.
:set nu	Muestra numeradas las líneas del texto.
:set nonu	Oculto la numeración de las líneas.
:set list	Despliega un carácter especial para marcar el final de la línea.
:set nolist	Oculto el carácter especial al final de las líneas.
G	Manda el cursor al final del archivo.
IG	Manda el cursor a la línea uno del archivo.
/cadena	Busca una cadena de donde está el cursor hacia abajo.
?cadena	Busca una cadena de donde está el cursor hacia arriba.
n	Muestra la siguiente cadena localizada.
N	Muestra la anterior cadena localizada.

3.18 Seguridad

El sistema operativo Solaris está diseñado para que los usuarios puedan acceder fácilmente a sus recursos y compartir información con otros usuarios. Este sistema también ha incluido características para preservarlo de usuarios no autorizados y para proteger los recursos de los usuarios, sin impedimento a los usuarios autorizados.

Existen diferentes niveles de seguridad y por lo tanto debe tenerse presente que la seguridad es relativa, dado que algunas de ellas proporcionan sólo una protección limitada y pueden ser evitadas por usuarios con conocimientos, muchas pueden ser atacadas con éxito por expertos.

Proporcionar seguridad que sea altamente resistente requiere procedimientos y técnicas especiales por lo que en este trabajo de tesis sólo se verán los fundamentos.

Archivos de contraseña

El sistema operativo Solaris guarda información acerca de los usuarios en dos archivos, `/etc/passwd` y `/etc/shadow`. Estos archivos son usados por el programa `login` para validar los usuarios y preparar el entorno de trabajo inicial.

El archivo `/etc/passwd`

El archivo `/etc/passwd` es parte integral de la seguridad del sistema, sin este archivo cualquiera podría entrar a él. Existe una línea en el archivo para cada usuario y para ciertos nombres de presentación utilizados por el sistema. Cada una de estas líneas contiene una secuencia de campos, separados por dos puntos que son características del usuario en particular. El siguiente ejemplo muestra un archivo `/etc/passwd` típico:

```
root:x:0:1:Super-user:/:bin/ksh
daemon:x:1:1::/
bin:x:2:2::usr/bin:
sys:x:3:3:/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
nobody:x:60001:60001:Nobody:/:
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS 4.x Nobody:/:
listen:x:901:401:Dave Listen:/export/home/listen:/bin/sh
```

El archivo `/etc/passwd` contiene las siguientes líneas:

```
loginID:x:UID:GID:comentario:directorío casa:login shell
```

Cada una de las partes anteriores contiene un significado en particular como se describe a continuación:

loginID Este campo es llamado "nombre de usuario" y contiene el nombre de presentación, que tiene de uno a ocho caracteres para los usuarios.

x El segundo campo contiene una letra x para proporcionar un grado de seguridad ya que representa el passwd del usuario.

UID El tercer campo contiene el identificador de usuario que es usado por el sistema operativo para identificarlo. Los números de usuario tienen un rango de 100 a 60000.

Los valores de 0 a 99 son reservados para los usuarios de sistema. El UID 60001 es reservado para el usuario nobody y el 60002 para el usuario noaccess.

Es posible duplicar los números de usuario, si dos usuarios tienen el mismo número podrán entrar a los mismos archivos.

GID El cuarto campo contiene el grupo identificador que es usado por el sistema para identificar al usuario del grupo primario. El rango de los grupos puede ser por lo general de 100 a 60000, los valores de 0 a 99 son reservados para el sistema.

comentario El quinto campo sirve para colocar comentarios sobre el usuario, generalmente contiene el nombre del usuario. El campo de comentarios para los nombres de presentación asociados con órdenes de sistema se utilizan para describir el propósito de la orden.

directorio casa El sexto campo contiene el nombre de la ruta para el directorio propio del usuario.

login shell El séptimo campo designa al programa que el sistema ejecuta automáticamente cuando el usuario abre una sesión. Éste se denomina shell de presentación, pudiendo tener ésta línea alguno de los siguientes: /bin/sh, /bin/csh ó /bin/ksh.

El archivo /etc/shadow

Existe una línea en el archivo /etc/shadow por cada línea del archivo /etc/passwd. El archivo /etc/shadow contiene información acerca de la contraseña de un usuario y datos referentes al envejecimiento de la contraseña. Solo el super-usuario puede tener acceso al archivo. A continuación se muestra el ejemplo de este archivo:

```
root:kdjSDF45opSD:6445::::::  
daemon:NP:6445::::::  
bin:NP:6445::::::  
sys:NP:6445::::::  
adm:NP:6445::::::  
lp:NP:6445::::::  
smtp:NP:6445::::::
```

uucp:NP:6445:.....
nuucp:NP:6445:.....
listen:*LK*.....
nobody:NP:6445:.....
noaccess:NP:6445:.....
nobody4:NP:6445:.....

El archivo /etc/shadow contiene las siguientes líneas:

loginID:password:lastchg:min:max:warm:inactive:expire:

Cada una de las partes anteriores contiene un significado en particular como se describe a continuación:

loginID El primer campo de la línea contiene el nombre de la presentación.

password El segundo campo contiene la contraseña cifrada para ese nombre de presentación. La contraseña cifrada consta de 13 caracteres del alfabeto de 64 caracteres, que incluye los siguientes: ., /, 0-9, A-Z y a-z. Es posible encontrar en este campo las letras NP (no password) cuando no existe contraseña para ese nombre de presentación, puede tener una x para los nombres de presentación uucp y nuucp y *LK* para la presentación listen.

lastchg El tercer campo indica el número de días entre el 1 de enero de 1970 y el día en que la contraseña fue modificada por última vez.

min El cuarto campo indica el número mínimo de días requerido entre cambios de la contraseña. Un usuario no puede cambiar su contraseña hasta que transcurra ese número de días.

max El quinto campo indica el número máximo de días que una contraseña es válida. Transcurrido ese número de días, el usuario tendrá que cambiar la contraseña.

warn El sexto campo indica el número de días antes de la expiración de una contraseña que el usuario es avisado. Se enviará un mensaje de aviso a un usuario cuando éste se presente para notificarle que su contraseña está por expirar dentro de esos días.

inactive El séptimo campo indica el número de días de inactividad permitido a este usuario. Si ese número de días transcurre sin que el usuario se presente, su línea de presentación se bloquea.

expire El octavo campo indica la fecha absoluta (especificada mediante el número de días desde el 1 de enero de 1970) a partir de la cual el nombre de presentación ya no puede ser utilizado.

El noveno campo es una opción que actualmente no es utilizada.

El archivo /etc/group

Todos los usuarios tienen un grupo primario especificado en el archivo /etc/passwd, el archivo /etc/group define todos los grupos del sistema y especifica los grupos secundarios a los que puede acceder un usuario que se encuentre incluido en éstos. Es importante mencionar que un usuario puede tener un grupo primario y hasta 15 grupos secundarios.

Cuando se ve el contenido del archivo /etc/passwd, se observa en el cuarto campo (GID) sólo un número que hace referencia al grupo al que pertenece el usuario; para conocer el nombre del grupo es necesario ver el archivo /etc/group que contiene además del ID del grupo, el nombre de éste.

El siguiente ejemplo es típico del archivo /etc/group cuando se instala el sistema operativo. Los grupos listados son definidos por la instalación del software. Para definir un nuevo grupo se utiliza el comando groupadd mismo que actualiza este archivo en la creación de un nuevo grupo.

```
root::0:root
other::1:
bin::2:root, bin, daemon
sys::3:root, bin, sys, adm
```

```
adm::4:root,adm,daemon
uucp::5:root,uucp
mail::6:root
tty::7:root, tty, adm
lp::8:root, lp, adm
nuucp::9:root, nuucp
staff::10:
daemon::12:root, daemon
sysadmin::14:lister, torey
nobody::60001:
noaccess::60002:
```

El archivo /etc/group contiene las siguientes líneas:

```
groupname:password:GID:userlist
```

Cada una de las partes anteriores contiene un significado en particular como se describe a continuación:

groupname El primer campo contiene el nombre del grupo que puede contener hasta ocho caracteres.

password El segundo campo contiene la contraseña cifrada del grupo aun que generalmente no contiene contraseña.

GID El tercer campo contiene el número identificador del grupo.

userlist El cuarto campo contiene una lista de usuarios separados por comas que hacen referencia a grupos secundarios a los que pertenece.

El archivo /var/adm/sulog

El archivo /var/adm/sulog guarda por default una bitácora de los usuarios que han usado el comando su para entrar al sistema como superusuario. El ejemplo siguiente muestra un archivo /var/adm/sulog típico:

more /var/adm/sulog

```
su 10/20 14:50 + console root-sys
su 10/20 16:50 + pts/2 lister-root
su 10/20 17:00 + pts/2 lister-root
su 10/21 11:06 + pts/2 lister-root
su 10/21 12:03 + pts/3 lister-root
su 10/24 13:00 + pts/3 lister-root
su 10/24 13:50 + console root-sys
su 10/28 08:14 + pts/3 lister-root
su 10/28 08:50 + console root-sys
su 11/02 11:07 - pts/3 rimmer-root
```

Cada uno de los campos del ejemplo anterior contiene un significado especial:

su	El primer campo despliega el comando su.
10/20	El segundo campo muestra el día en que el usuario se conectó como root.
14:50	El tercer campo muestra la hora en que el usuario entró al sistema como root.
+ ó -	El cuarto campo muestra un signo (+) si fue exitoso el intento de conectarse como usuario root, de lo contrario aparecerá un signo negativo (-).
console	El quinto campo lista el nombre del terminal desde donde se ejecutó el comando su.
root-sys	El sexto campo lista el nombre del usuario y el nombre del usuario al que se conectó.

El archivo /var/adm/sulog puede ser de gran ayuda cuando deseamos identificar a los usuarios que han entrado al sistema con privilegios mayores o menores de los que tienen.

El archivo /etc/default/passwd

El archivo /etc/default/passwd contiene variables importantes para la definición de algunos parámetros relacionados con el programa passwd. Dentro de este archivo existen las variables MAXWEEKS, MINWEEKS y PASSELENGTH que asignan ciertos valores por omisión a las contraseñas.

El contenido típico del archivo `/etc/default/passwd` es el siguiente:

```
# more passwd

#ident "@ (#) passwd.dfl 1.3 92/07/14 SMI"
MAXWEEKS=
MINWEEKS=
PASSENGTH=6
```

La variable `MAXWEEKS` especifica el valor del número máximo de semanas que es válido el password antes de tener que ser cambiado por todos los usuarios. Si es definido un valor nulo (por default), solo los usuarios que tienen un valor de `MAXWEEKS` especificado en el archivo `/etc/shadow` tendrán que cambiar su password en el tiempo especificado.

La variable `MINWEEKS` especifica el número mínimo de semanas que debe durar un password antes de poder ser cambiado por los usuarios. Si se define un valor nulo, solo los usuarios que tienen un valor de `MINWEEKS` especificado en el archivo `/etc/shadow` son limitados a poder cambiar su password hasta que transcurra el número mínimo de semanas.

La variable `PASSENGTH` especifica un valor mínimo para la longitud del password, para todos los usuarios se especifica un valor entre 6 y 8 caracteres.

El archivo `/etc/default/login`

El archivo `/etc/default/login` ayuda a restringir el acceso a la cuenta de superusuario. Contiene muchas variables pero hay dos de relevante importancia, la variable `PASSREQ` y la variable `CONSOLE`

Si el valor de la variable `PASSREQ` es colocado con "YES" (que es su valor por default) provoca que los usuarios que no se les asignó un password al momento de crear sus cuentas les pida que introduzcan un password en la primera vez que desean entrar al sistema.

La variable `CONSOLE` puede ser usada para especificar una de tres condiciones para conectarse al sistema como superusuario. Si el valor de la variable `CONSOLE` es definido como `/dev/console`, el acceso al sistema como superusuario únicamente podrá realizarse desde la consola del sistema. Si se intenta conectarse como superusuario desde cualquier otra parte que no sea la consola mandará el siguiente mensaje de error:

```
# rlogin CMR
Not on system console
Connection closed.
```

Si la variable no es definida, el acceso a la cuenta de superusuario puede hacerse desde cualquier parte de la red del sistema.

Si el valor de la variable es definido como nulo (`CONSOLE=`), el acceso a la cuenta de superusuario no es permitido. En este caso, solo podrán tenerse los privilegios de superusuario entrando al sistema como usuario y de ahí utilizar el comando "su" para escalar a superusuario.

Estas condiciones son proporcionadas por el sistema para obtener una mayor seguridad en el sistema.

El archivo `/etc/default/su`

El archivo `/etc/default/su` define los accesos de todos los usuarios hechos con el comando "su". Contiene muchas variables pero hay dos de relevante importancia, la variable `SULOG` y la variable `CONSOLE`.

El valor de la variable `SULOG` especifica el nombre del archivo en el cual todos los intentos de acceso a otros usuarios son registrados. Si el valor de la variable es indefinido no serán registrados los intentos de acceso a otros usuarios.

La variable `CONSOLE` por default es ignorada porque se encuentra comentada con el símbolo (#) en el archivo, pero aún así los intentos de acceso a otros usuarios serán enviados por medio de un mensaje a la consola.

Si el valor de la variable `CONSOLE` es definido como `/dev/console` (sin estar comentada), todos los intentos de acceso exitoso a la cuenta de superusuario serán mandados a la consola y al archivo de registro `/var/adm/sulog`, si no son exitosos los intentos de acceso únicamente se registrarán en el archivo.

El archivo `/etc/host.equiv` y el archivo `.rhosts`

Al nivel anfitrión, cada sistema en la red contiene un archivo llamado `/etc/host.equiv`. Este archivo contiene una lista de las máquinas que son de "confianza" para ese anfitrión. Los usuarios en máquinas remotas listadas en este archivo pueden abrir una sesión de trabajo remotamente sin suministrar una contraseña. Un ejemplo del archivo `/etc/host.equiv` es el siguiente:

```
# more /etc/host.equiv
CABD      root
CMR       root
CEGR      sop
```

Si el archivo contiene una línea con sólo un signo más (+), esta máquina confía en todos los usuarios remotos.

Existe otra facilidad para forzar la seguridad a nivel usuario. Un usuario que tenga un directorio propio en una máquina remota puede tener un archivo de nombre `.rhosts` en el directorio propio de usuario en esa máquina.

Este archivo se utiliza para permitir o denegar acceso al nombre de presentación de ese usuario, dependiendo de en qué máquina y qué usuario esté tratando de obtener acceso a esa presentación. El archivo `.rhosts` define usuarios "equivalentes", a los que se proporcionan los mismos privilegios de acceso

Una entrada en el archivo `.rhosts` es o bien un nombre de máquina, indicando que este usuario es de confianza cuando accede al sistema desde la máquina especificada, o un nombre de máquina seguido de un nombre de presentación, indicando que el nombre de presentación listado es de confianza cuando accede al sistema desde la máquina especificada. Un ejemplo típico de este archivo es el siguiente:

```
# more .rhosts
CABD      root
CMR       root
```

Permisos de archivos

Existen tres clases de permisos para archivos que corresponden con las clases de usuario: el propietario (o usuario) del archivo, el grupo al que pertenece el propietario y los otros usuarios del sistema. Por ejemplo al listar el archivo `dia` con el comando `ls` y la opción `-l` obtenemos lo siguiente:

```
# ls -l dia
-rwxr-xr-x 1 you group1 362 Feb 2 08:08 dia
```

Las tres primeras letras del campo de permisos, como se ve en la salida de `ls -l` anterior, hacen referencia a los permisos del propietario, las tres siguientes a los miembros del grupo del propietario y las tres últimas a los otros usuarios.

En la entrada correspondiente al archivo denominado `dia` en el ejemplo anterior, las tres primeras letras, `rwx`, muestran que el propietario del archivo puede leerlo (`r`), escribirlo (`w`) y ejecutarlo (`x`).

El segundo grupo de tres letras, `r-x`, indica que los miembros del grupo pueden leer y ejecutar el archivo, pero no pueden escribirlo. Los tres últimos caracteres, `r-x`, muestran que todos los demás pueden leer y ejecutar el archivo, pero no escribirlo.

Si se tienen permisos de lectura de un archivo, se puede visualizar su contenido. El permiso de escritura significa que puede alterar su contenido. El permiso de ejecución significa que puede ejecutar el archivo como un programa.

Con respecto a los directorios, el permiso de lectura permite listar el contenido del directorio, el permiso de escritura permite crear o eliminar archivos o directorios dentro de ese directorio, y el permiso de ejecución permite moverse a ese directorio con la orden `cd` o utilizarlo como parte en un nombre de camino.

El comando chmod

En algún momento se puede desear que los directorios o archivos tengan diferentes permisos para que sean públicos, es decir, permitir que sean accedidos por cualquier persona. Otros archivos tendrán información no compartida fuera del grupo de trabajo o incluso otros serán privados y se querrá que nadie vea su contenido.

El sistema operativo Solaris permite fijar los permisos de cada uno de los archivos. Sólo el propietario de un archivo puede modificar los permisos, para permitir o impedir la lectura, escritura o ejecución por él mismo, su grupo u otros usuarios.

Para cambiar los permisos de un archivo se utiliza el comando chmod (change mode). Con chmod se especifica en primer lugar los permisos que se van a cambiar: u para usuarios, g para el grupo, o para otros. En segundo lugar se especifica cómo se cambian: + (añadir permisos) o - (quitar permisos) para leer, escribir o ejecutar. En tercer lugar se especifica el archivo al que hacen referencia los cambios. Por ejemplo:

```
# ls -ld dias
```

```
drwxr-xr-x 1 you group1 362 Feb 2 08:08 dias
```

```
chmod go-rx dias
```

```
# ls -l dias
```

```
drwx----- 1 you group1 362 Feb 2 08:08 dias
```

En el ejemplo anterior se solicitó en forma extendida el listado del directorio dias, se cambiaron sus permisos utilizando el comando chmod y se listó nuevamente para ver los nuevos permisos. El comando chmod en el ejemplo anterior eliminó (-) los permisos de lectura y de ejecución (rx) al grupo y a otros (go) correspondientes al directorio dias.

También se pueden añadir permisos con la orden chmod, por ejemplo:

```
chmod ugo+rwx dias
```

```
# ls -l dias
```

```
drwxrwxrwx 1 you group1 362 Feb 2 08:08 dias
```

La orden anterior añadió (+) permisos de lectura, escritura y ejecución (rwx) para el usuario, el grupo y otros (ugo) en el directorio dias.

La forma del comando chmod descrita anteriormente utilizando la notación ugo+/-rwx, permite cambiar los permisos en relación a su valor actual. Como propietario del archivo, puede añadir o quitar permisos según se desee. Otra forma de la orden chmod permite fijar los permisos directamente, utilizando un código numérico (octal) para especificarlas.

Este código representa el permiso de un archivo mediante tres números: uno para los permisos del propietario, otro para los permisos del grupo y otro para los permisos del resto. Estos tres dígitos aparecen juntos como un número de tres dígitos. Por ejemplo, la siguiente orden fija los permisos de lectura, escritura y ejecución para el propietario, no permitiendo a ningún otro hacer nada con el archivo:

```
# chmod 700 días
```

Para ver cómo se representan los permisos con este código, se muestra la siguiente tabla:

	Propietario	Grupo	Otros
Read	4	0	0
Write	2	0	0
Execute	1	0	0
Suma	7	0	0

La tabla anterior muestra cómo 700 representa los permisos de días. Cada columna de la tabla hace referencia a uno de los usuarios (propietario, grupo u otros). Si un usuario tiene permiso de lectura, se suma 4; para fijar el permiso de escritura, se suma 2, y para fijar el permiso de ejecución, se suma 1. La suma de los números de cada columna es el código de permiso del usuario.

En la siguiente tabla se muestra cómo la orden `chmod 750` fija el permiso de lectura, escritura y ejecución para el propietario, y los permisos de lectura y ejecución para el grupo:

```
# chmod 750 días
```

	Propietario	Grupo	Otros
Read	4	4	0
Write	2	0	0
Execute	1	1	0
Suma	7	5	0

Se puede utilizar la orden `chmod` para fijar los permisos de forma relativa o absoluta de cualquier archivo del que sea propietario.

El comando `umask`

La orden `chmod` permite alterar los permisos archivo por archivo. La orden `umask` permite hacer esto automáticamente cuando se crea cualquier archivo o directorio. Todos los usuarios tienen un `umask` por omisión definido en su `profile`.

`umask` permite especificar los permisos de todos los archivos que se crean después de emitir la orden `umask`. En lugar de tratar con permisos sobre archivos individuales, se determinan con una sola orden los permisos de todos los archivos futuros.

Para especificar umask se utiliza un código numérico para representar permisos absolutos de la misma forma que chmod. Por ejemplo, 777 significa permisos de lectura, escritura y ejecución para el propietario, el grupo y los otros (rwxrwxrwx).

Se especifica los permisos diciéndole a umask que reste de los permisos completos el valor 777. por ejemplo, después de emitir la orden siguiente, todos los nuevos archivos de esta sesión tomarán los permisos rwxr-xr-x:

```
# umask 022
```

Para ver como funciona el ejemplo anterior, se debe de tomar en cuenta que corresponde a un valor numérico de 755, y 755 es el resultado de restar la mascara 022 en este ejemplo de 777.

Para asegurarse que sólo el dueño pueda leer, escribir o ejecutar sus archivos, se deberá emitir la orden umask al comienzo de la sesión colocando la siguiente línea en el archivo .profile:

```
# umask 077
```

La orden anterior equivale a utilizar chmod 700 o chmod go-rwx, pero umask se aplica a todos los archivos que se creen en la sesión actual después de emitir la orden.

El bit sticky

El bit sticky es un permiso especial para un determinado archivo o directorio. Poner el bit sticky tiene efectos importantes cuando se pone en un directorio ya que proporciona una seguridad añadida. Existen algunos directorios en el sistema que deben permitir permisos generales de lectura, escritura y búsqueda por ejemplo, tmp y spool. Un peligro con estos es que otros podrían borrar archivos de usuario. El bit de sticky puede ponerse a directorios para evitar que otros eliminen los archivos de un usuario. Si se pone el bit de sticky a un directorio, los archivos en ese directorio pueden eliminarse si una o más de las siguientes condiciones es verdad:

- El usuario es el propietario del archivo.
- El usuario es el propietario del directorio.
- El usuario tiene permiso de escritura para el archivo.
- El usuario es el superusuario.

Para poner el bit de sticky se utiliza el comando chmod, por ejemplo:

```
# chmod 1753 dias  
o  
# chmod +t dias
```

Si es colocado el bit sticky en un archivo, aparecerá una t en la parte de ejecución del campo de permisos de otros.

Niveles de seguridad de sistema operativo

El Departamento de Defensa de los Estados Unidos ha producido estándares para diferentes niveles de seguridad de sistemas informáticos. Estos estándares han sido publicados en el documento Trusted Computer System Evaluation Criteria, conocido como "Libro Naranja".

Hay siete niveles de seguridad informática descritos en el Libro Naranja. Estos niveles están organizados en cuatro grupos, A, B, C y D, de exigencias de seguridad decrecientes. Dentro de cada división hay uno o más niveles de seguridad, etiquetados con números. Desde el nivel superior de seguridad hasta el inferior, estos niveles son A1, B3, B2, B1, C2, C1 y D. Todos los requerimientos de seguridad para un nivel inferior también son válidos para los niveles superiores, de modo que todo requerimiento de seguridad para un sistema B1 es también una exigencia para un sistema B2, B3 o A1.

Protección mínima (clase D)

Los sistemas con una clasificación de clase D tienen características mínimas de protección. Un sistema no tiene que pasar ninguna prueba para ser clasificado como clase D.

Protección de seguridad discrecional (clase C1)

Para que un sistema tenga el nivel C1, debe proporcionar una separación entre usuarios y datos. Los controles discrecionales necesitan estar disponibles para permitir a un usuario limitar el acceso a datos. Los usuarios deben ser identificados y validados.

Protección de acceso controlado (clase C2)

Para que un sistema tenga un nivel C2, un usuario debe ser capaz de proteger los datos de modo que esté disponible sólo a usuarios específicos. Debe mantener una auditoría que lleve la cuenta de los accesos a objetos, como archivos. La seguridad C2 también exige que no haya datos disponibles como residuo de un proceso, de modo que los datos generados por el proceso en memoria temporal o registros sean borrados.

Protección de seguridad etiquetada (clase B1)

Los sistemas en el nivel de seguridad B1 deben tener capacidades de control de acceso obligatorias. En particular, los sujetos y objetos que son controlados deben ser etiquetados individualmente con un nivel de seguridad. Las etiquetas deben incluir niveles de seguridad jerárquicos, como "sin clasificar", "secreto" y "alto secreto" y categorías (tales como nombres de grupo o equipo). El control de acceso discrecional también debe estar presente.

Protección estructurada (clase B2)

Para que un sistema satisfaga el nivel B2 de seguridad, debe existir un modelo de seguridad formal. Los canales de cobertura, que son canales no usados normalmente para comunicaciones sino que pueden ser usados para transmitir datos, deben estar restringidos. Debe haber un diseño de alto nivel verificable y la comprobación debe confirmar que este diseño ha sido implementado. Debe designarse un oficial de seguridad que implemente políticas de control de acceso, mientras que el administrador del sistema usual debe limitarse a las funciones necesarias para el funcionamiento del sistema.

Dominios de seguridad (clase B3)

La seguridad de los sistemas en el nivel B3 debe basarse en un modelo conceptualmente sencillo, pero completo. Debe haber un argumento convincente, pero no una prueba formal, de que el sistema implementa el diseño. Debe incluirse la capacidad de especificar protección de acceso para cada objeto y de especificar sujetos permitidos, junto con el acceso permitido a cada uno de ellos, y sujetos no permitidos. Un monitor de referencia que tome las peticiones de acceso de los usuarios y apruebe o desapruebe el acceso basado en políticas de control de acceso debe ser implementado. El sistema debe ser altamente resistente a la penetración y la seguridad debe ser a toda prueba. Debe proporcionarse una facilidad de auditoría que pueda detectar violaciones de seguridad potenciales.

Diseño verificado (clase A1)

Las capacidades de un sistema de clase A1 son idénticas a las del sistema de clase B3. Sin embargo, el modelo formal para un sistema de clase A1 debe ser verificado formalmente como seguro.

3.19 Adición de una impresora

El sistema operativo Solaris incluye una colección de programas, denominados el sistema lp, para imprimir archivos y documentos. Se puede utilizar para imprimir cualquier cosa, desde archivos de texto simple hasta grandes documentos con formatos complejos. Proporciona una interfaz simple y uniforme a una amplia variedad de impresoras, que van desde las máquinas de matrices de puntos hasta las impresoras láser.

Cuando están instaladas las utilidades de impresora de línea (lp), existe generalmente un guión de shell que está dispuesto para iniciar el planificador lp cuando la computadora entra en estado multiusuario. Cuando se añade una impresora, es necesario detener el planificador lp, identificar la impresora, reiniciar el planificador, decir que la impresora está preparada para aceptar trabajos y activar la impresora.

En el ejemplo siguiente se añadirá una impresora de matriz de puntos y se colocará al puerto 11 (dev/term/11).

Una vez que se ha conectado la impresora al puerto de la computadora se deberá asegurar que el puerto tiene los permisos correctos, la propiedad de usuario (lp) y la propiedad de grupo (bin) apropiados para impresión. Como primer paso se deberá hacer lo siguiente:

```
# chown lp /dev/term/11
# chgrp bin /dev/term/11
# chmod 600 /dev/term/11
# ls -l /dev/term/11
```

```
crw----- 1 lp bin 1, 0 oct 27 13:59 /dev/term/11
```

Para desactivar el planificador lp se deberá teclear:

```
# /usr/sbin/lpshut
Print services stopped.
```

Para identificar la impresora al sistema lp se tecleará:

```
# lpadmin -p Data -I /usr/lib/lp/model/standard -l /dev/term/11
```

Esto le dirá el nombre de Data a la impresora, utilizará el archivo /usr/lib/lp/model/standard para la definición del interfaz para la impresora e identificará el puerto 11 como el puerto al que está conectada. Después de esto se deberá reiniciar el planificador lp tecleando:

```
# /usr/lib/lpsched
Print services started.
```

Se deberá permitir que la impresora acepte peticiones lp tecleando:

```
# accept Data
Destination "Data" now accepting requests.
```

Capacite luego la impresora tecleando:

```
# enable Data
Printer "Data" now enables.
```

La impresora deberá estar ahora disponible para impresión. Se puede comprobar imprimiendo un archivo de texto.

Es importante mencionar que la adición de una impresora al sistema por medio de línea de comandos es un trabajo complicado ya que requiere de amplios conocimientos acerca del sistema, este trabajo de tesis está realizado para aquellas personas que se inician en la administración básica del sistema operativo por lo que se deja la tarea de dar una mayor información del tema a la administración avanzada.

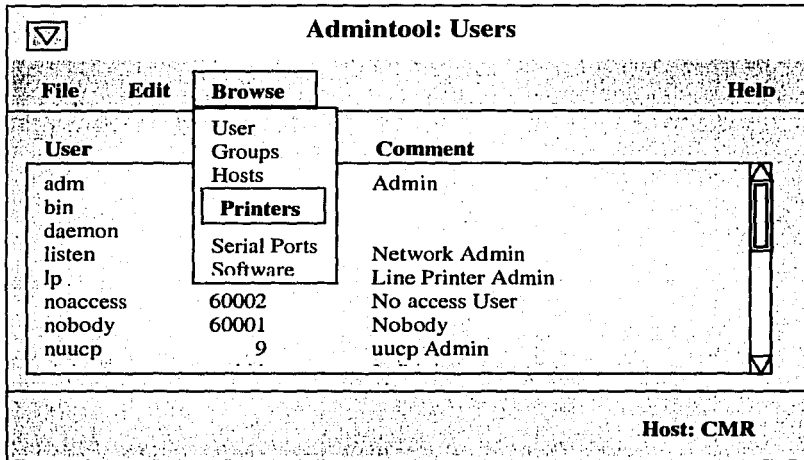
Existe otra forma más fácil de adicionar una impresora al sistema por medio de la interfaz gráfica llamada admintool, esta herramienta permite una mayor facilidad para adicionar usuarios, grupos de trabajo e impresoras.

En el siguiente ejemplo se verá como adicionar una impresora por medio de la herramienta admintool.

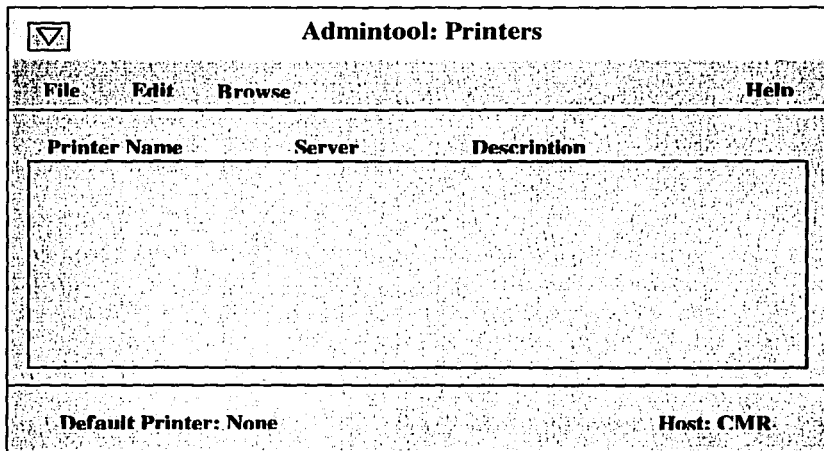
Como primer paso se deberá llamar a la interfaz admintool, para poder hacer esto, se deberá ser miembro del grupo sysadmin o ser superusuario.

admintool &

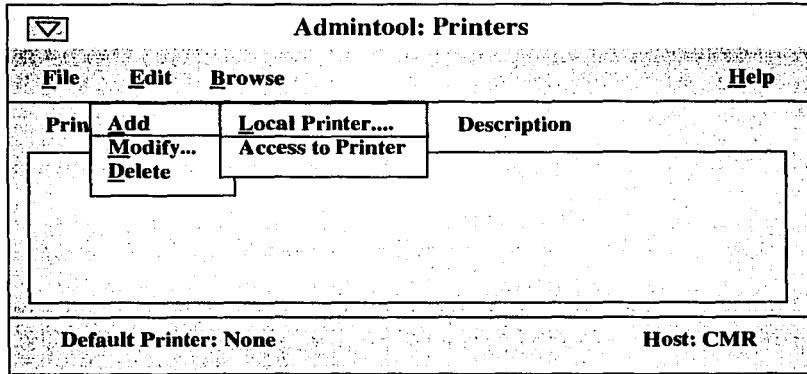
Una vez introducida la orden anterior aparecerá la siguiente ventana, seleccionando la palabra Browse y escogiendo la palabra printers.



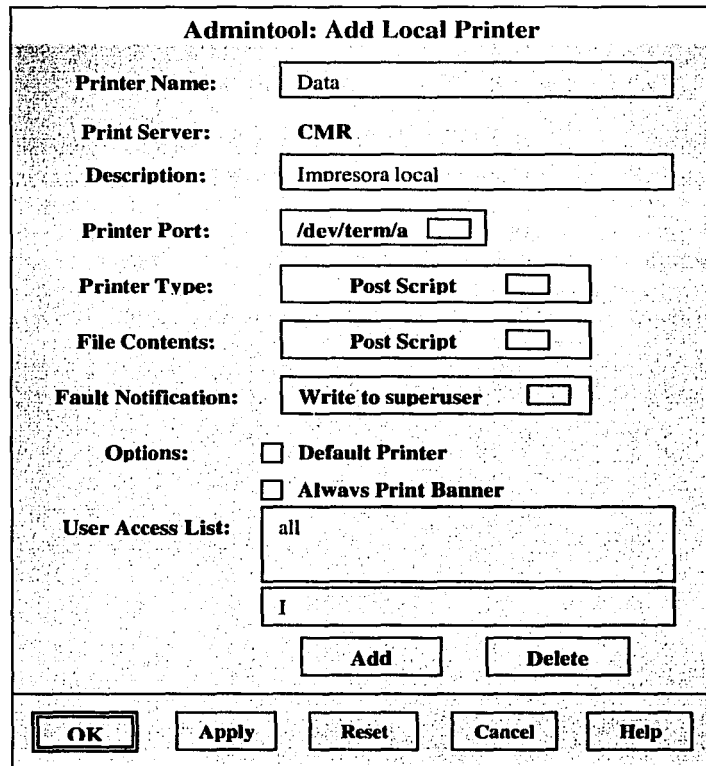
Una vez seleccionada la opción printers aparecerá la siguiente ventana:



En la ventana anterior se deberá seleccionar de la barra de menú la opción Edit, Add y Local printer, como se ve en el siguiente ejemplo:



Una vez seleccionado la opción de impresora local aparecerá la siguiente ventana donde será insertada la información necesaria para adicionar la impresora al sistema.



En la ventana anterior debe ser insertada la información solicitada para la adición de la impresora, tomando en cuenta que el número máximo de caracteres en el nombre es de 14 y que el comentario opcional (descripción) sirve para identificar más fácilmente la impresora cuando se tienen varias dentro del sistema.

Es necesario seleccionar las opciones solicitadas por los menús de la ventana, como el nombre del puerto que puede ser `/dev/bpp0`, `dev/term/a`, o `/dev/term/b` para esa opción, o si se desea que siempre aparezca el banner cuando se mande algún archivo a imprimir y escribir la lista de los usuarios que pueden utilizar la impresora.

También es posible adicionar una impresora remota por medio de la herramienta `admintool`, el ejemplo siguiente adicionará una impresora remota al sistema.

Para este punto, el primer paso será llamar a la interfaz `admintool` y seleccionar del menú la opción `Browse e impresoras`, aparecerá la ventana de impresoras y del menú `Edit` se seleccionaran las opciones `Add` y después `Access to printer`. Al seleccionar estas opciones aparecerá la siguiente ventana:

Admintool: Add Access To Printer

Print client: **CMR**

Printer Name: data

Print Server: cabd

Print Server: postscript printer

Option Default Printer

OK Apply Reset Cancel Help

En la ventana anterior se deberá introducir la siguiente información para adicionar la impresora:

- El nombre de la impresora.
- El nombre del servidor de impresión.
- Un comentario opcional que servirá para identificar la impresora.
- Seleccionar si esta impresora será por default para el sistema.
- Presionar el botón `OK` para adicionar la impresora y desaparecer la ventana.

TESIS CON
FALLA DE ORIGEN

Una vez que el proceso anterior ha sido terminado, la interfaz admintool desplegará la siguiente ventana con la lista de las impresoras disponibles.

Admintool: Printers			
File	Edit	Browse	Help
Printer Name	Server	Description	
data	cabd	postscript printer	
matriz	cmr	postscript printer	
Default Printer: None		Host: CMR	

Para eliminar una impresora del sistema bastará con llamar a la interfaz admintool, seleccionar la impresora y del menú Edit elegir la opción Delete, esto borrará la impresora.

Comandos básicos de lp

Una vez que se ha instalado una impresora al sistema o si ya existe alguna puede ser necesario llevar a cabo algunas tareas para su administración, para ello existen varias órdenes, en la siguiente tabla se muestra los comandos más usados de lp para administrar las impresoras:

Nombre del comando	Descripción
lp	Manda a imprimir un archivo.
lpstat	Despliega el status del servidor de impresión.
cancel	Cancela las peticiones de impresión.
lpadmin	Configura y mantiene el servicio de impresión.
accept	Permite encolar peticiones de impresión.
reject	Rechaza ordenes de impresión si está ocupada.
lpmove	Mueve ordenes de impresión.
enable	Activa las peticiones de impresión lp.
disable	Desactiva las peticiones de impresión lp.

El comando lp

La orden básica para imprimir un archivo es lp (line printer). El mensaje de confirmación de lp devuelve un "request id" que se puede utilizar para comprobar el estado del trabajo o para cancelarlo de ser necesario. El ID presentado por el sistema lp se compone de dos partes, el nombre de la impresora y el número que identifica la petición particular, por ejemplo:

```
# lp archivo1
```

```
request id is Data-170 (1 file)
```

El formato del comando es:

```
lp [-opción] nombre de archivo(s)
```

Si se desea imprimir un archivo en una impresora diferente a la de nuestro sistema o indicarle al sistema lp que desean varios juegos de copias del mismo documento, se pueden utilizar varias opciones con el comando lp para realizar tareas específicas.

Opciones del comando lp:

- d La opción -d (destination), seguida por el nombre de una impresora que no es la determinada por default permite indicarle al sistema lp que se desea imprimir en esa impresora.
- n La opción -n permite informarle al sistema lp el número de copias que se desean del archivo.
- o La opción -o le indica la sistema lp que no se desea que imprima la página banner.
- m La opción -m le informa al sistema lp que sea enviada una notificación cuando se imprime el trabajo.

A continuación se presentan ejemplos del comando lp con cada una de las opciones mencionadas:

```
# lp archivo1  
request id is Data-17 (1 file)
```

```
# lp -n 2 archivo1  
request id is Data-21 (1 file)
```

```
# lp -d laser archivo1  
request id is Data-25 (1 file)
```

```
# lp -d laser -o nobanner archivo1  
request id is Data-31 (1 file)
```

```
# lp -m -n 4 archivo1  
request id is Data-23 (1 file)
```

El comando lpstat

Debido a que los trabajos de impresión no se realizan inmediatamente, y puesto que pueden ser enviados a impresoras localizadas fuera del área de trabajo, se necesita una manera de comprobar el estado de los trabajos de impresión. La orden lpstat proporciona esta posibilidad, junto con otra información útil, como las impresoras que están disponibles en el sistema y el número de trabajos enviados, por ejemplo:

```
# lpstat
```

```
data -57 jmf Nov 26 00:29
```

En el ejemplo anterior se muestra que el trabajo está listo para impresión, pero que aún no ha comenzado.

El formato del comando es:

```
lpstat [-opciones]
```

El comando lpstat puede ser utilizado con opciones para indicarle alguna tarea específica.

Opciones del comando lp:

- a La opción -a muestra qué impresoras están en espera de una petición de impresión.
- d La opción -d despliega el nombre de la impresora por omisión.
- s La opción -s muestra cuál es la impresora del sistema y en cuáles se puede trabajar.
- t La opción -t presenta un estado general del sistema de impresión (qué impresoras están disponibles y cuántos trabajos tiene cada una).
- o La opción -o proporciona las salidas de las peticiones de impresión.
- p La opción -p Despliega información detallada de las impresoras del sistema.

Un ejemplo de cómo se usa el comando lpstat para determinar la impresora por omisión del sistema es el siguiente:

```
# lpstat -s
scheduler is running
system default destination: laser
system for bashful: data
system for laser: data1
```

El comando cancel

Hay ocasiones en las que se necesita cancelar un trabajo de impresión. La orden cancel permite detener cualquiera de los trabajos de impresión, incluso los que se están imprimiendo. Si lp da el ID deskjet-98 a un trabajo de impresión y se desea pararlo, puede utilizarse la siguiente cadena para borrarlo del sistema de impresión:

```
# cancel deskjet - 98
```

```
request "deskjet - 98" canceled
```

Puede suceder que no se haya anotado el número de trabajo cuando se envió a imprimir, en este caso se puede utilizar el comando lpstat para obtenerlo.

También es posible cancelar un trabajo de impresión con la opción -u, el nombre de usuario y el nombre de la impresora, por ejemplo:

```
# cancel -u sop deskjet
```

```
cmr: deskjet-5 dequeued
```

3.20 File systems

Para los usuarios del sistema operativo Solaris un file system es una colección de archivos y directorios usados para colocar y organizar información. Para el sistema operativo un file system es una colección de estructuras de control y bloques de datos que ocupan un espacio definido por una partición.

Existen varios tipos de file systems que pueden ser implementados en discos duros, CD-ROM o diskette:

- ufs: El file system de UNIX. El ambiente del sistema Solaris por default utiliza file systems basados en la BSD (Berkeley Software Distribution), ufs es designado para optimizar el trabajo en discos por medio de grupos de cilindros y bloques continuos de datos.
- hfs: El file system de High Sierra. Es un file system de solo lectura para CD-ROM basado en la norma ISO 9660 para cubrir las normas de estandarización.
- pcfs: El file system de PC: Permite leer y escribir archivos en un disco con sistema operativo DOS.
- RAM-based: Este file system también es llamado pseudo file system, su mayor característica es que no reside en un dispositivo de hardware, solo reside en memoria física cuando el sistema operativo está corriendo.

El ambiente del sistema operativo Solaris coloca los datos en archivos jerárquicos de forma lógica, los cuales se encuentran dentro de los file systems. El término file system describe una jerarquía de archivos y directorios en la partición, el término estructura de archivos describe las entradas al árbol de archivos.

La estructura de archivos de árbol consiste en el file system de root y la colección de file systems desmontables del sistema.

Para entender más la estructura del file system ufs se darán algunas definiciones de términos importantes.

Raw y partición de bloque

Una partición en crudo es un espacio de disco a la cual se le asigna un tamaño en particular y un nombre pero que no es un file system. Una partición puede ser usada como un dispositivo crudo, tal como el área de swap. Una partición de bloque es un file system, éstas dos particiones tienen nombres ya definidos usando los directorios llamados /dev/rdisk y /dev/dsk.

La etiqueta del disco (VTOC)

La etiqueta del disco contiene la tabla de particiones para el disco y hace referencia al volumen de la tabla de contenido (VTOC). Una partición de disco puede contener un file system que el sistema operativo Solaris interpreta como una organización de directorios y archivos.

La etiqueta del disco es localizada en el primer sector del disco de 512-bytes.

El Boot block

El programa bootstrap también llamado bootblock se encuentra a partir del sector 15 del disco, solo el file system de root tiene activo el boot block, aun que el espacio para éste se encuentre al inicio de cada file system.

El superblock

El file system es descrito por el superblock, que empieza a partir del sector 16 del disco después del bootblock. El superblock es una tabla de información acerca del file system que incluye el número de bloques de datos, el número de los grupos de cilindros, el tamaño de los bloques de datos, el nombre del punto de montaje, una descripción del hardware (derivada para la etiqueta) y el estado del file system: limpio, estable o activo.

Respaldo del superblock

Dado que el superblock contiene datos críticos, éste es replicado en cada grupo de cilindros para protección en caso de que sea dañado, esto es hecho cuando el file system es creado.

En la siguiente figura se muestra la estructura del file system ufs.

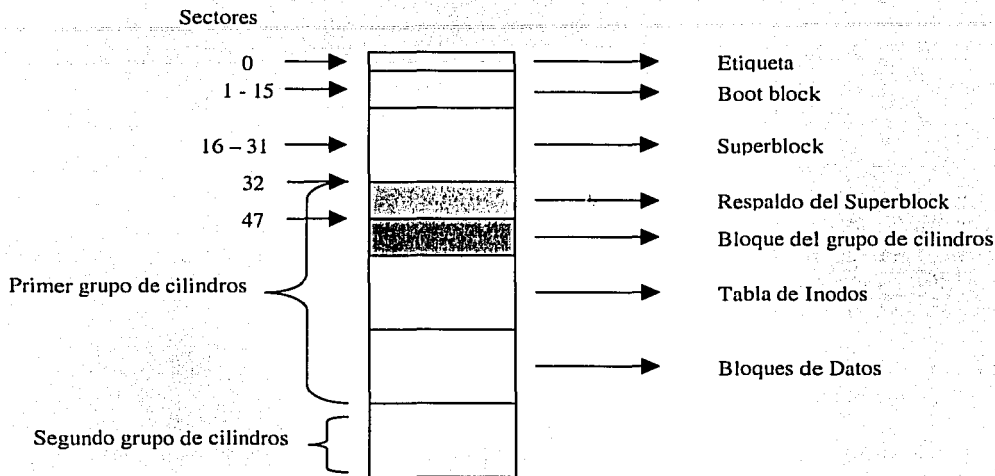


Fig. 3.46 Estructura del file system ufs.

Grupo de cilindros

La partición es dividida en grupos de cilindros para mejorar el acceso al disco, por omisión es de 16 cilindros por grupo. El file system constantemente optimiza el disco para colocar los archivos de datos en un grupo de cilindros, reduciendo así el movimiento de la cabeza, el file system colocará los archivos a través de varios grupos de cilindros si es necesario.

El bloque de los grupos de cilindros es una tabla que describe al grupo de cilindros; dentro de los datos que contiene están el número de inodos, el número de bloques de datos en el grupo de cilindros, el número de directorios y los bloques libres, los inodos libres, los fragmentos libres en el grupo del cilindro, el mapa de bloques libres y el mapa de inodos usados.

La tabla de inodos contiene los inodos para el grupo de cilindros, un inodo es la descripción interna para el archivo y la localidad en el bloque de datos. Cada grupo de cilindros contiene una porción del total de los inodos.

El bloque de datos es la unidad para colocar los datos en el file system del sistema operativo, por omisión el tamaño del bloque de datos es de 8192 bytes.

Cualquier partición en un disco que se va a particionar necesita tener un file system creado antes de adicionar datos. Para adicionar un nuevo file system se puede utilizar el comando newfs o mkfs.

El ejemplo siguiente muestra cómo adicionar un file system en la primera partición de un disco:

```
# newfs /dev/rdisk/c0t2d0s0
```

```
newfs: construct a new file system /dev/rdisk/c0t2d0s0: (y/n)? y
/dev/rdisk/c0t2d0s0: 41040 sectors in 57 cylinders of 9 tracks, 80 sectors
21.0 MB in 4 cyl groups (16 c/g, 5.90MB/g, 2688 i/g)
super-block backups (for fsck -F ufs -o b=#) at:32, 11632, 23232, 34832,
```

En el ejemplo anterior como primer paso para la creación del nuevo file system se le introdujo la cadena con la orden, una vez introducida pregunta si se quiere construir un nuevo file system en la partición señalada, aquí es importante estar seguros de lo que se desea hacer, ya que el comando sobrescribe en los datos si ya existe el file system, para continuar con la creación se responderá a la pregunta afirmativamente.

El comando del ejemplo anterior creará el directorio lost+found que es usado por el comando fsck para verificar y reparar el file system en caso de ser necesario.

El comando newfs usa una escala para colocar el porcentaje mínimo de espacio libre para mantener el file system. Los valores pueden estar en el rango de 1 a 10 %, esto puede ser ajustado usando la opción -m con el porcentaje de espacio libre durante la creación del file system. Adicionalmente el comando tune2fs -m % libre puede ser usado para cambiar el valor del porcentaje mínimo para un file system sin montar.

El comando fsck

El comando (fsck) file system check usa parámetros conocidos e información redundante para verificar el file system de inconsistencias, cuando el sistema operativo es iniciado, el programa fsck verifica el estado de cada file system.

El programa fsck corre en dos modos: modo interactivo y no interactivo.

El sistema operativo corre el programa fsck en modo no interactivo durante el levantamiento del sistema. En este modo, el programa fsck solo corrige problemas conocidos que pueden solucionarse perfectamente sin la intervención de un operador. Si el programa detecta una inconsistencia termina con un mensaje y se sale, en este caso el administrador del sistema tendrá que correr el programa fsck en forma interactiva.

Cuando se corre el programa fsck en forma interactiva lista cada problema encontrado seguido por una sugerencia de acción correctiva y el administrador del sistema decidirá si se realiza o no.

Para detectar inconsistencias en los file systems el programa fsck verifica en orden los siguientes puntos:

El resumen de información del superblock incluyendo:

- El tamaño del file system.
- El número de inodos.
- El contador de bloques de datos libres.
- El contador de inodos libres.

El bloque de grupos de cilindros:

- Bloque de datos libres ocupado por archivos.
- Contador de bloques de datos correctos.
- Contador de inodos libres correctos.

Información de inodos:

- El contador de ligas incorrectas en los inodos, los cuales indican malas entradas a los directorios.
- Inconsistencias entre el valor del tamaño del inodo y el número de bloques de datos que tienen referencia.
- Los inodos asignados y no asignados.

Información del bloque de datos del file system:

- Un bloque de datos que pertenece a varios file systems.
- Bloques de datos marcados como libres pero en uso.
- Bloques de datos marcados como en uso pero libres.

Información del directorio:

- El número de inodos no asignados en los directorios.

La siguiente es la salida del programa fsck cuando no se registran inconsistencias en los file systems:

```
# fsck /dev/rdisk/c0t3d0s7

** /dev/rdisk/c0t3d0s7
** Last Mounted on /export/home
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
2 file, 9 used, 21060 free (14 frags, 21699 blocks, 0.1 % fragmentation)
#
```

En el ejemplo anterior las primeras líneas indican los pasos que el programa fsck realizó, la última línea lista los valores del file system:

- El número de archivos usados (dos archivos).
- El número de Kbytes usados (9 usados)
- El número de Kbytes libres (21699 libres).
- El reacomodo del espacio libre dentro de los bloques libres y fragmentos de bloques.
- La porción de bloques de fragmentos libres en porcentaje.

Algo que es importante tomar en cuenta es que nunca se debe de correr el programa fsck con el file system ocupado ya que éste podría reportar errores y posiblemente borre archivos de usuarios cuando intenten leer o escribir información.

Es recomendable que el programa fsck sea corrido en modo simple usuario o que el file system esté desmontado, tomando en cuenta que el file system de root (/) y /usr no pueden desmontarse del sistema.

Si el superblock se daña puede causar problemas con los file systems, dado que la información contenida en el superblock es vital para verificar el file system, el programa fsck puede ser utilizado para usar un bloque del respaldo del superblock con la opción -o y con b igual al número de la bandera, cuando la b es usada, la localidad del respaldo del superblock tendrá que ser provista.

El respaldo del superblock se encuentra a partir del sector 32 del disco, si este bloque también está dañado, existen más bloques de respaldo. El comando newfs con la opción -N puede ser usado para determinar las localidades de los demás bloques.

Para utilizar el respaldo del superblock localizado en el sector 32 para el file system se deberán seguir los siguientes pasos:

1.- Listar el respaldo del superblock para la partición.

```
# newfs -N /dev/rdsk/c0t0d0s7
```

2.- Seleccionar uno de los bloques y correr el programa fsck de la siguiente manera.

```
# fsck -o b=32 /dev/rdsk/c0t0d0s7
```

Con los dos pasos anteriores podrá restaurarse el file system, pero hay que tomar en consideración que existen varios bloques de respaldo y que en algún momento habría que probar con cada uno de ellos.

Cuando se necesite monitorear el estado de los file systems, los comandos df, du y quot ayudan al administrador a monitorear el uso de los file systems incluyendo el espacio en disco utilizado y libre, el tamaño de los archivos y directorios y el número de bloques pertenecientes a varios usuarios.

El comando `df` es usado para desplegar información para cada file system montado, el formato del comando es:

`df [-k] [directorio]`

La opción `-k` despliega el uso en kbytes y sustrae el espacio reservado por el sistema operativo para la cantidad de espacio disponible.

El siguiente ejemplo muestra la manera de desplegar la capacidad de los file systems:

`df -k`

Filesystem	kbytes	used	avail	capacity	Mounted on
/dev/dsk/c0t3d0s0	38111	19196	18877	51%	/
/dev/dsk/c0t3d0s6	565503	361529	203409	64%	/usr
/proc	0	0	0	0%	/proc
fd	0	0	0	0%	/dev/fd
/dev/dsk/c0t3d0s1	25159	4886	20248	20%	/var
/dev/dsk/c0t3d0s5	27439	20362	7050	75%	/opt
swap	45980	12	45968	1%	/tmp

En el ejemplo anterior cada columna tiene un significado:

Filesystem: El file system montado.

kbytes: El tamaño del file system en Kbytes (1024 bytes).

used: El número de Kbytes usado.

avail: El número de Kbytes disponibles.

capacity: El espacio del file system usado en porcentaje.

Mounted on: El punto de montaje.

Otro comando para monitorear el estado de los file systems es el `du`, éste es usado para desplegar el número de bloques (512 bytes) por archivos y directorios en el disco. El formato del comando es:

`du [-a] [-s] [-k] [directorio]`

Cada una de las opciones para el comando `du` vistas anteriormente sirven para presentar los datos de forma específica, el significado de las opciones es el siguiente:

`-k` Despliega el espacio ocupado en Kbytes.

`-s` Despliega el resumen de la información en bloques de 512 bytes. Usando la opción `-k` junto con la opción `-s` mostrará la información en Kbytes.

- a Despliega el número de bloques usado por cada uno de los archivos y directorios en forma jerárquica contenidos dentro del directorio especificado.

A continuación se muestran ejemplos de las opciones del comando `du` mencionadas anteriormente para que sea más fácil entender las salidas de la orden.

Kbytes

```
# cd /opt
du -k
8      ./lost+found
3      ./SUNWits/Graphics-sw/xil/lib
4      ./SUNWits/Graphics-sw/xil
16     ./SUNWits/Graphics-sw/xgl/demo
...
...
38     ./netscape/movemail-scr
11392 ./netscape
20362 .
```

Resumen

```
# cd /usr
# du -s /usr
723098 /usr
```

Bloques

```
# du -a /usr
16     /usr/lost+found
2      /usr/x
2      /usr/lib/libICE.so
2      /usr/lib/libICE.so.6
...
6      /usr/kvm
...
724590 /usr
```

Otro comando para monitorear el estado de los file system es el `quot` que permite desplegar la información del espacio ocupado en el disco (en Kbytes) por los usuarios, el formato del comando es:

```
quot [-af] [file sistem]
```

Las opciones que se utilizan con el comando `quot` significan lo siguiente:

- a Ejecuta el comando en todos los file systems montados.
- f Incluye una columna con el número de archivos perteneciente a cada usuario.

Es posible que en algún momento se necesite saber la cantidad de espacio que están ocupando los usuario dentro de los file systems, el siguiente ejemplo muestra la salida del comando `quot`:

```
# quot -af
```

```
/dev/rdisk/c0t3d0s0 (/):
 14326      1284      root
 4798       37        bin
 31         27        lp
 31         14        uucp
 1          1         adm
 1          1         sys
/dev/rdisk/c0t3d0s6 (/usr):
 197459     6964      root
 162389    11884     bin
 2149      232      lp
 798       24       uucp
 1         1        adm
```

En el ejemplo anterior cada una de las columnas tiene un significado, la primera columna representa los Kbytes usados, la segunda el número de archivos y la tercera especifica el nombre del dueño de los archivos.

Montado de los file systems

Montar es el proceso por el cual los file system son separados y adheridos a una estructura jerárquica (estructura de árbol), los file system son colocados en directorios llamados puntos de montaje.

El montaje y desmontaje de los file systems ocurre durante la iniciación y el apagado del sistema basado en el archivo `/etc/vfstab`, (virtual file system table).

Es frecuente para el administrador del sistema encontrar situaciones que requieran montar o desmontar los file systems, al realizar respaldos, verificar los file systems para encontrar inconsistencias o para repararlos.

En la versión de Solaris 2.7 los file systems pueden tener un tamaño igual a 2 Gbytes, siendo el máximo de un terabyte.

Siempre que una operación de montaje es realizada (por ejemplo, el comando `mount` o `umount`), la información con referencia al montaje es grabada en el archivo `/etc/mnttab`. El comando `mount`, sin opciones, es usado para obtener información de todos los archivos montados del sistema.

el siguiente ejemplo muestra el uso del comando mount sin argumentos:

mount

```
/ on /dev/dsk/c0t3d0s0 read/write/setuid/largefiles/logging on Mon Nov 18
17:58:09 2000
/usr on /dev/dsk/c0t3d0s6 read/write/setuid/largefiles/logging on Mon Nov 18
17:58:09 2000
/proc on /proc read/write/setuid on Mon Nov 18 17:58:09 2000
/dev/fd on fd read/write/setuid on Mon Nov 18 17:58:09 2000
/var on /dev/dsk/c0t3d0s1 nolargefiles/read/write/setuid on Mon Nov 18 17:58:09 2000
/opt on /dev/dsk/c0t3d0s5 setuid/read/write/largefiles on Mon Nov 18 17:58:09 2000
/tmp on swap read/write on Mon Nov 18 17:58:09 2000
/pns on dbserver:/export/pse soft/nosuid/remote on Mon Nov 18 17:58:09 2000
```

Los campos desplegados por el comando mount son:

- Campo 1: Hace referencia al punto de montaje (directorio), usado para adherir el file system listado en el campo 3.
- Campo 2: La palabra "on".
- Campo 3: El archivo de bloques de dispositivo especial que hace referencia al file system que se encuentra disponible o montado.
- Campo 4: La lista de opciones de montaje (por ejemplo, largefiles o nolargefiles, read o read/write, setuid o nosuid). Los file system logging también son indicados en este campo.
- Campo 5: La palabra "on".
- Campo 6: La fecha y la hora en que el file system fue montado.

En la figura 3.47 se muestra el file system de root.

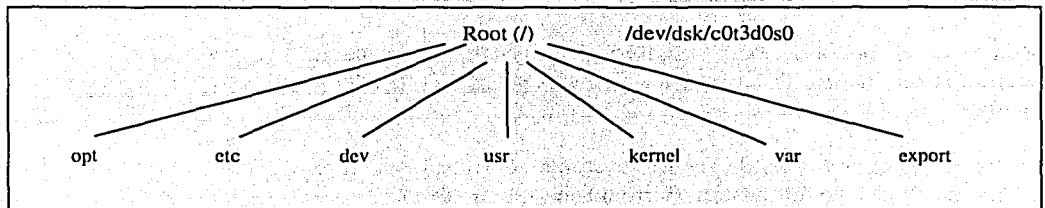


Fig. 3.47 file system de root

Para montar file systems manualmente se puede usar el comando mount sin opciones, por ejemplo:

```
# mount /dev/dsk/c0t3d0s7 /export/home
```

Existen opciones que permiten adicionar algunas propiedades especiales al file system como la propiedad UFS logging que no es habilitada por default. Para habilitarla sobre un file system se debe especificar la opción `-o logging` con el comando `mount` cuando este será montado, por ejemplo:

```
# mount -o logging /dev/dsk/c0t3d0s6 /usr
```

La opción anterior permite montar el file system con propiedades de logging (permite realizar transacciones).

La opción `nolargefiles` para el comando `mount` obliga a todos los archivos subsecuentes que se escriban dentro del file system a ser más pequeños de 2 Gbytes, por ejemplo:

```
# mount -o nolargefiles /dev/dsk/c0t3d0s7 /export/home
```

Usando la opción `nolargefile` fallará el montaje si el file system contiene archivos largos al tiempo del montaje y el file system ya no contiene archivos largos pero no ha sido corrido el programa `fsck`.

Es importante mencionar que siempre que un file system contenga uno o mas archivos iguales o más grandes de 2 Gbytes deberá ser montado con la opción `largefiles`, esto no puede ser remontado con la opción `nolargefiles` sin que antes sean movidos a otro file system y haberle corrido el programa `fsck`. Puede ser montado un file system usando el contenido del archivo `/etc/vfstab`, si los recursos del file system son listados en el archivo, se puede especificar el montaje del file system de la siguiente manera:

```
# mount /export/home
```

El archivo vfstab

El archivo `/etc/vfstab` (virtual file system table) contiene las entradas por omisión para el montaje de los file systems al inicio del sistema, el formato del archivo es un registro por línea, siete campos por registro con un guión indicando un valor nulo para ese campo.

#device #to mount	device to fsck	mount point	FS type	fsck pass	mount at boot	mount ops
#/dev/dsk/c1d0s2	/dev/rdisk/c1d0s2	/usr	ufs	1	yes	-
/proc	-	/proc	proc	-	no	-
fd	-	/dev/fd	fd	-	no	-
swap	-	/tmp	tmpfs	-	yes	-
/dev/dsk/c0t3d0s0	/dev/rdisk/c0t3d0s0	/	ufs	1	no	-
/dev/dsk/c0t3d0s6	/dev/rdisk/c0t3d0s6	/usr	ufs	1	no	logging
/dev/dsk/c0t3d0s3	/dev/rdisk/c0t3d0s3	/export	ufs	5	yes	logging
/dev/dsk/c0t3d0s7	/dev/rdisk/c0t3d0s7	/export/home	ufs	5	yes	logging
/dev/dsk/c0t3d0s5	/dev/rdisk/c0t3d0s5	/opt	ufs	8	yes	logging
/dev/dsk/c0t3d0s1	-	-	swap	-	no	-

El significado de las columnas en el archivo `/etc/vfstab` es:

Device to mount: Identifica el dispositivo especial de bloque para un file system UFS local (por ejemplo `/dev/dsk/c0t3d0s0`), el nombre del recurso para un file system de red (por ejemplo `dbserver:/export/pse`), el nombre del bloque de dispositivo especial para swap (por ejemplo `/dev/dsk/c0t3dos1`), o el directorio `/proc` y el tipo de file system `proc`.

Device to fsck: Identifica el nombre de dispositivo especial en crudo que corresponde al file system identificado por el campo *device to mount* (por ejemplo `/dev/rdisk/c0t3d0s0`). Esto determina la interfaz cruda para el programa `fsck`.

FS type: La identificación del tipo de file system para el campo del dispositivo a montar. Este es siempre `ufs` para los recursos del archivo local.

Mount at boot: Si se encuentra un "yes" o un "no" se indica cuando el recurso puede o no ser montado cuando el sistema entra en el nivel 2 o cuando el comando `mountall` es usado.

Existen tres excepciones que son `root (/)`, `/usr` y `/var`; si `/var` es un file system separado. Todos éstos son montados por el comando `mount` especificado en el script `/etc/rcS.d/S30rootusr.sh`.

Mount options: Indica una lista de opciones separadas por comas para un file system local UFS, estas opciones pueden ser: `read/write/setuid/largefiles`.

Durante el inicio o el apagado del sistema, todos los file systems son montados o desmontados automáticamente. El comando para montar todos los file systems locales es:

```
# mountall -l
```

La opción `-l` indica los file systems locales. El comando `mountall` lee la información del archivo `/etc/vfstab` bajo la columna *mount at boot* y monta todos los file systems con un valor en la columna de *yes* cuando el sistema inicia.

Para desmontar todos los file systems locales se utiliza el comando:

```
# umountall -l
```

El desmontaje de un file system hace a los archivos inaccesibles para los usuarios. Para desmontar un file system se puede hacer de dos formas, por ejemplo:

```
# umount /dev/dsk/c0t2d0s5
```

El ejemplo anterior utiliza el nombre del dispositivo especial del file system.

```
# umount /database
```

Como en el ejemplo anterior el file system puede ser desmontado usando el punto de montaje.

Es importante mencionar que un file system ocupado (incluyendo cualquier usuario que contenga cualquier directorio dentro del file system) no podrá ser desmontado.

Una vez que un nuevo disco ha sido colocado en el sistema y los file system aun no han sido creados, puede hacerse manualmente, para ello se deberán seguir los siguientes pasos:

1.- Crear el punto de montaje para el file system, construyendo un nuevo directorio donde será colocado.

```
# mkdir /database
```

2.- Montar el file system dentro del punto de montaje.

```
# mount /dev/dsk/c0t2d0s5 /database
```

3.- Para que el file system sea montado automáticamente cuando el sistema inicia se deberá editar la entrada en el archivo /etc/vfstab.

3.21 Administración de discos

La administración de discos dentro del sistema operativo Solaris es un proceso que requiere de amplios conocimientos del sistema, en este tema de tesis se muestran los puntos más usados que deben tomarse en cuenta para llevar a cabo una buena administración básica de estos dispositivos. Para entender mas fácilmente las tareas que se llevan a cabo con éstos, se presentan los siguientes términos.

Sector: Es la unidad más pequeña en un plato de disco. Un sector contiene 512 bytes y estos son sinónimos de los bloques de disco.

Tracks: Los tracks son una serie de sectores. Los sectores que conforman un track pueden ser leídos o escritos por una cabeza dada en una posición durante una revolución sencilla de disco (una vuelta). Las superficies son formateadas como tracks de datos concéntricos, las cuales son completadas en una rotación de disco completa por una posición de cabeza dada.

Cilindro: Es la suma de tracks provistos por todas las cabezas en una posición dada.

Para hacer la escritura o lectura a un disco, éstos son divididos en platos, a su vez los platos se dividen en sectores y se agrupan dentro de cilindros los cuales son leídos por cabezas flotantes, puesto que estas realizan la lectura más eficiente cuando los datos se encuentran almacenados en un mismo cilindro.

En los sistemas Sun, el almacenamiento en disco es dividido en secciones llamadas particiones (slices). Las particiones son agrupadas en cilindros que posibilitan al administrador para organizar funcionalmente los datos.

Por ejemplo, es importante que un file system y los programas importantes de sistema sean colocados en una partición, como los archivos asociados con root, o los programas generados por usuarios que deberán ser colocados en otra partición.

Un manejador de disco provisto por Sun puede contener ocho particiones, etiquetadas de la 0 a la 7, la partición 0 por default contiene los archivos de root, por omisión la partición 2 representa el disco entero, el área de swap puede ser otra partición dependiendo de la selección al momento de la instalación.

Nombres de Dispositivos

En el ambiente del sistema operativo Solaris, los dispositivos son referenciados de cuatro formas diferentes: Lógica, Física, Instancia y BSD.

Los nombres de dispositivos lógicos son usados por el administrador del sistema para hacer referencia a los dispositivos. Estos nombres son ligados simbólicamente a los correspondientes nombres de dispositivo físicos. Los nombres lógicos son localizados en el directorio /dev y son creados al mismo tiempo que los nombres físicos, por ejemplo:

```
# ls /dev/dsk
c0t0d0s0    c0t0d0s4    c0t3d0s0    c0t3d0s4    c0t6d0s0
c0t0d0s1    c0t0d0s5    c0t3d0s1    c0t3d0s5    c0t6d0s1
```

Los sistemas Sun utilizan la siguiente convención de nombres para describir el nombre del dispositivo lógicamente para un disco conectado a un controlador SCSI:

```
/dev/[r]dsk/c#t#d#s#
c# Indica el número de controladora.
t# Indica el número target.
d# Indica el número de la unidad lógica (LUN).
s# Indica el número de partición.
```

Para entender mejor cómo funciona la asignación del nombre lógico ver la figura 3.48

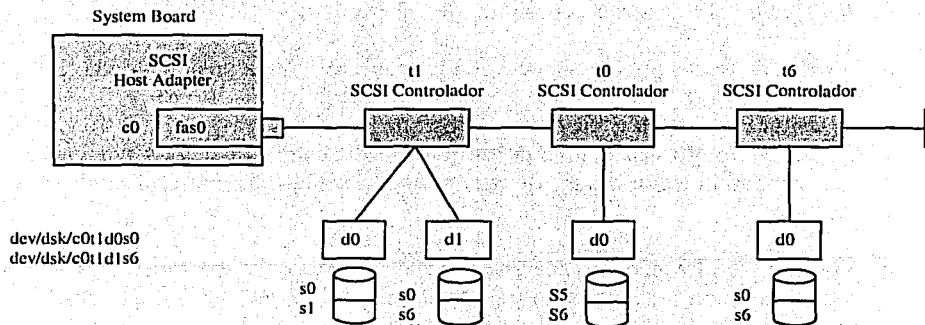


Fig. 3.48 Nombre de dispositivo lógico.

Número de controladora: La controladora o interfaz puede contener números como c0, c1, c2, que son asignados automáticamente en secuencia al bus de la interfaz. Cualquier manejador de disco conectado al host SCSI tiene una dirección de dispositivo que empieza con c0, el controlador c1 corresponde al segundo adaptador SCSI.

El número de target: Los números de target tales como t0, t1, y t3 corresponden a la configuración del switch de dirección que es elegido para cada dispositivo. Una unidad de disco externo usualmente tiene un switch de dirección que está localizado en el panel posterior.

El número de disco: El número de disco es también conocido como el Logical Unit Number (LUN). Este número refleja el número de discos en la ubicación del target. Para cualquier disco SCSI fijado hay solo un disco en la dirección del target, de aquí que el número de disco es siempre configurado como d0.

La partición: El número de la partición puede contener un rango de 0 a 7. Los dispositivos de disco son accedidos por sus nombres de dispositivo lógico, y estos nombres deben incluir el número de partición.

El nombre lógico de dispositivo de disco es usado por los administradores del sistema (y usuarios) cuando usan comandos relacionados al file system y el disco.

Los nombres de dispositivo físico representan el nombre de la ruta de dispositivo completo de la jerarquía de información de dispositivos (o árbol). Los nombres físicos están debajo del directorio /devices donde son creadas las entradas durante la instalación o en una configuración subsecuente de dispositivos. El archivo /device provee un apuntador al kernel para los programas.

El objeto superior de la jerarquía es llamado el nodo root del árbol de dispositivos. Un objeto intermedio debajo del nodo root tiene un controlador de dispositivo asociado con el y es llamado una hoja.

El kernel identifica la ubicación física de dispositivos asociando un nodo con una dirección, *nodename@address* lo cual es llamado el nombre de dispositivo físico, por ejemplo, sd@1.

Los nombres de instancia son los nombres abreviados del kernel para todo posible dispositivo en el sistema. El comando dmesg despliega los nombres de instancia tales como sd0 y sd1.

Un nombre de dispositivo de disco por instancia es una abreviación para el nombre de dispositivo físico.

Un ejemplo de nombre de instancia es: sdn, donde s es igual a SCSI, d igual a disco y n es igual a número de disco lógico, tal como sd0 para el primer dispositivo de disco SCSI.

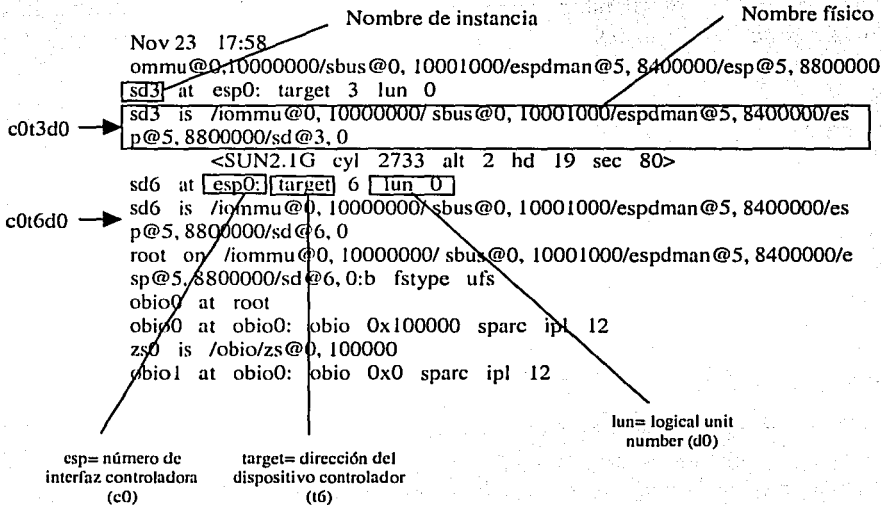
El comando dmesg

El comando `dmesg` identifica los dispositivos conectados al sistema usando nombres de instancia y nombres de dispositivo físico. Desde esta salida, es posible determinar los nombres de disco lógico.

El comando `dmesg` busca en un buffer del sistema mensajes de diagnóstico recientemente impresos y los imprime en la salida estándar. Los mensajes son aquellos impresos por el sistema cuando ocurre un error. Si está dada la bandera “-“ entonces `dmesg` computa en incrementos los mensajes nuevos desde la última vez que fue corrido y los coloca en la salida estándar.

Un ejemplo de la salida del comando `dmesg` es la siguiente:

```
# dmesg | more
```



El comando format

La salida del comando `format` despliega nombres de dispositivo tanto lógicos como físicos, como se muestra en el siguiente ejemplo:

```
# format
```

```
Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:
```

```
0. c0t3d0 <SUN424 cyl 1151 alt 2 hd 9 sec 80>
```

```
/iommu@f,e0000000/sbus@f,e0001000/espdma@f,400000/esp@f,800000/sd@3,0
```

```
Specify disk (enter its number):
```

El ejemplo de `format` identifica un dispositivo de disco SCSI (`sd@3, 0`) conectado al adaptador de host SCSI (`esp@f, 80000`), el cual está conectado al controlador de acceso directo a memoria (DMA) (`espdma@f, 40000`), el que está conectado a la interfaz `sbus` (`sbus@f, e000100`).

El comando `format` tiene más aplicaciones, pero en éste capítulo solo se menciona para conocer las salidas de los nombres de dispositivos que proporciona, más adelante se mencionará mayor información del comando.

Reconfiguración de dispositivos

Cuando son añadidos dispositivos nuevos al sistema, es realizado un arranque de reconfiguración para forzar al sistema a reconocer los nuevos dispositivos. Este proceso crea un nuevo árbol de información de dispositivos y directorios `/devices` y `/dev`.

Los siguientes son los pasos para reconfigurar un sistema de forma tal que reconozca un dispositivo nuevo. Este ejemplo ilustra la adición de un disco nuevo al sistema.

1.- Crear un archivo `/reconfigure`, así el sistema realizará una reconfiguración cuando sea arrancado.

```
# touch /reconfigure
# init 5
```

2.- Configurar el número de target SCSI en el disco nuevo a uno que no esté en uso.

3.- Conectar el disco nuevo al sistema.

4.- Encender la máquina.

5.- Arrancar el sistema a nivel PROM

```
# ok boot
```

Después de configurar el sistema para reconocer el disco se puede empezar el proceso de configuración de particiones del disco.

Si un archivo `/reconfigure` no se creó antes de apagar la máquina se usa el comando `boot -r` a nivel PROM para forzar el sistema a reconfigurar.

El comando `drvconfig` también puede ser usado para actualizar el directorio `/devices`. Éste es normalmente ejecutado después de que un controlador nuevo ha sido instalado.

Particiones de disco y el comando format

Durante la instalación, la utilería installtool crea las particiones y las carga junto con el sistema operativo. Después de la instalación, la utilería format permite crear nuevas particiones o modificar las existentes.

Es importante mencionar que si se reparticiona un disco existente después de usar la utilería installtool, los datos en estos file systems podrían ser perdidos. Es necesario respaldar los file systems antes de reparticionar el disco. El comando format es una herramienta de mantenimiento de disco que es usada desde el shell de la ventana o desde un CD-ROM de instalación.

Las razones básicas para reparticionar un disco son:

- Volver a dimensionar el disco.
- Volver a etiquetar un disco con una nueva tabla de partición (VTOC).

Los tamaños de las particiones pueden ser configurados manualmente por el comando format o editando el archivo /etc/format.dat.

Conforme a Solaris 2.3, el comando format configura y etiqueta automáticamente controladores SCSI, éstos no necesitan estar definidos en el archivo /etc/format.dat

La utilería format permitirá crear particiones traslapadas. Si se traslapan particiones no se sabrá hasta que se intente usar los comandos newfs y mount. Cuando se intente entrar datos a la partición tendrá un mensaje de error relativo al file system.

Una etiqueta de disco también llamada tabla de volumen del disco (VTOC) contiene:

- Tablas de partición del disco.
- Nombre de volumen opcional que identifica el dispositivo de disco.
- Tags de partición opcionales que nombran los puntos de montaje estándar para cada una de las particiones. Los tags son de valor limitado pues no son usados por el sistema operativo y están limitados en su definición.
- Banderas de partición opcionales que etiquetan si cada partición es escribible o montable. Las banderas de partición son de valor limitado.

La etiqueta de disco ocupa el primer sector de un disco. Fig. 3.49

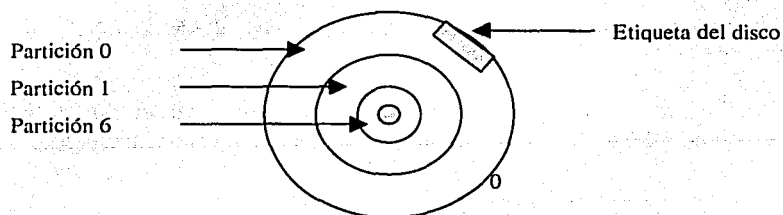


Fig. 3.49 Partición de disco y etiqueta.

La tabla de partición primaria define los límites de cada partición y el número de cilindros en la partición, como se muestra en el siguiente ejemplo:

Current partition table (SUN0424):

Part	Tag	Flag	Cylinders	Size	
0	root	wm	0-45	16.17 MB	(46/0/0)
1	swap	wu	46-136	31.99 MB	(91/0/0)
2	backup	wm	0-1150	404.65 MB	(1151/0/0)
3	unassigned	wm	0	0	(0/0/0)
4	unassigned	wm	0	0	(0/0/0)
5	unassigned	wm	0	0	(0/0/0)
6	usr	wm	137-1150	356.48 MB	(1014/0/0)
7	unassigned	wm	0	0	(0/0/0)

Los límites de la partición deben marcar el comienzo y el final de todos los cilindros.

Las particiones son descritas por un offset (distancia desde el cilindro 0) y el tamaño (Fig. 3.50). Los offsets y el tamaño de las particiones para un disco son definidas en la tabla de partición.

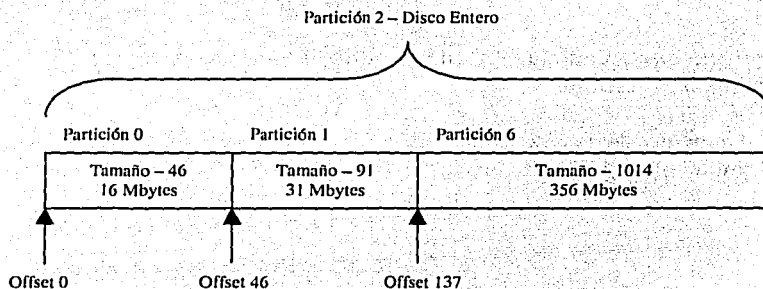


Fig. 3.50 Offsets y tamaños para las particiones de disco.

En la figura anterior el offset para la partición 0 es desde el cilindro 0 y tiene un tamaño de 46 cilindros. Por lo que la partición empieza en el cilindro 0 y termina en el cilindro 45. para la partición 1 el offset es desde el cilindro 46 y el tamaño es de 91 por lo que termina en el cilindro 136, y así sucesivamente.

Por omisión, para la partición 2 describe el disco entero.

En el tamaño de las particiones pueden darse dos condiciones no deseadas o de error:

- Espacio en disco desperdiciado debido a una mala partición que deja huecos entre particiones.

El decremento del tamaño de la partición puede dejar espacio desperdiciado entre una y la siguiente partición, como se ilustra en la siguiente figura.

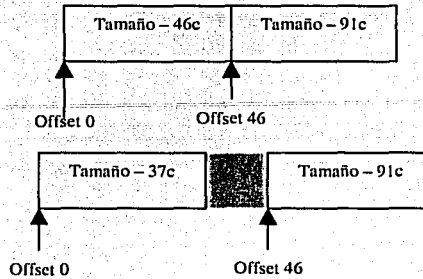


Fig. 3.51 Partición con espacio desperdiciado.

- Espacio en disco encimado debido a que las partición se crean una ocupando espacio asignado a otra partición.

Incrementar el tamaño de la partición puede crear que las particiones sean encimadas como se muestra a continuación.

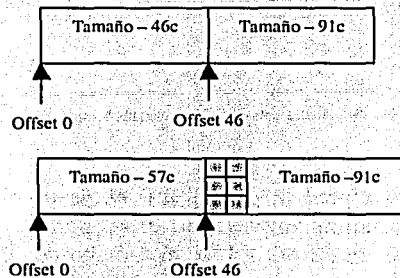


Fig. 3.52 partición con espacio encimado.

Por las razones anteriores es necesario tener cuidado al momento de modificar las particiones en el disco.

Los siguientes pasos describen cómo cambiar el tamaño de una partición.

1.- Escribir en el prompt format y presionar la tecla Enter.

format

Searching for disks...done

AVAILABLE DISK SELECTIONS:

0. c0t3d0 <SUN0424 cyl 1151 alt 2 hd 9 sec 80>
/iommu@f, e0000000/sbus@f, e0001000/espdma@f, 400000/esp@f, 800000/sd@3, 0

1. c0t0d0 <SUN0424 cyl 1151 alt 2 hd 9 sec 80>
/iommu@f, e0000000/sbus@f, e0001000/espdma@f, 400000/esp@f, 800000/sd@0, 0

2. c0t2d0 <SUN1.05 cyl 2036 alt 2 hd 14 sec 72>
/iommu@f, e0000000/sbus@f, e0001000/espdma@f, 400000/esp@f, 800000/sd@2, 0

Specify disk (enter its number): 2

Cuando el comando format es iniciado, cualquier disco disponible es descrito por el nombre lógico, el nombre de la marca del proveedor, los parámetros físicos y el nombre físico.

El nombre del proveedor usualmente empieza con SUNnnn. En el ejemplo anterior, el proveedor es SUN0424.

2.- Escoger el disco seleccionándolo por el número de la izquierda correspondiente a la descripción previa.

Es importante mencionar que si se cambia la geometría de la partición de cualquier file system existe el riesgo de dañarlo, esto puede suceder si se reduce el tamaño de la partición, se mueve el punto de inicio o se crea otra partición encimándola en una ya existente.

Selecting c0t2d0

[disk formatted]

FORMAT MENU:

disk	- select a disk
type	- select (define) a disk type
partition	- select (define) a partition table
current	- describe the current disk
format	- format and analyze the disk
repair	- repair a defective sector
label	- write label to the disk
analyze	- surface analysis
defect	- defect list management
backup	- search for backup labels
verify	- read and display labels
save	- save new disk/partition definitions
inquiry	- show vendor, product and revision
volname	- set 8-character volume name
!<cmd>	- execute <cmd>, then return
quit	

Una vez seleccionado el disco, el comando `format` despliega el menú principal. Las funciones primarias del menú del comando `format` que son descritas en este capítulo son:

`partition`: Despliega el menú de la partición.

`label`: Escribe la definición de la partición actual a la etiqueta del disco.

`veryfi`: Lee y despliega la etiqueta del disco.

3.- Escribir en el prompt de `format partition`.

```
format> partition
PARTITION MENU:
  0 - change "0" partition
  1 - change "1" partition
  2 - change "2" partition
  3 - change "3" partition
  4 - change "4" partition
  5 - change "5" partition
  6 - change "6" partition
  7 - change "7" partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cdm> - execute <cdm>, then return
quit
```

El menú de la partición es desplegado. Este menú permite ejecutar las siguientes funciones:

- 0-7 Especifica el comienzo y el tamaño de las siete particiones.
- `select`: Selecciona una tabla de partición definida.
- `modify`: Modifica la tabla de partición definida.
- `name`: Le da nombre a la tabla de partición actual.
- `print`: Despliega la tabla de partición actual.
- `label`: Escribe la tabla actual a la etiqueta del disco (VTOC).

4.- Escribir en el prompt de `partition print`.

```
partition> print
Current partition table (original):
Total disk cylinders available: 2036 + 2 (reserved cylinders)
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 1830	901.20MB	(1831/0/0)
1	swap	wu	1831 - 1983	75.30MB	(153/0/0)
2	backup	wm	0 - 2035	1002.09MB	(2036/0/0)
3	unassigned	wm	0	0	(0/0/0)
4	unassigned	wm	0	0	(0/0/0)
5	unassigned	wm	0	0	(0/0/0)
6	unassigned	wm	0	0	(0/0/0)
7	home	wm	1984 - 2035	25.10MB	(51/0/0)

El nombre de la tabla de partición es desplegado en paréntesis en la primera línea de cada tabla.

Las columnas de la tabla tienen el siguiente significado:

- ❖ Part: Indica el número de la partición.
- ❖ Tag: El nombre de la partición.
- ❖ Flag: Indica la bandera de la partición.
- ❖ Cylinders: Muestra el rango de cilindros ocupados por cada partición.
- ❖ Size: Muestra el tamaño de la partición en Mbytes.
- ❖ Blocks: Muestra el tamaño de la partición en notación de cylinder/track/sector.

Si el disco tiene un nombre de volumen, puede ser listada la palabra Volume después de la primera línea de la tabla.

5.- Escribe 0 para cambiar la partición de root.

```
partition> 0
Part      Tag      Flag      Cylinders      Size      Blocks
0         root     wm        0 - 1830      901.20MB  (1831/0/0)
```

Enter partition id tag[root]:

Enter partition permission flags[wm]:

Enter new starting cyl[0]:

Enter partition size[1845648b, 1831c, 901.20mb]: 400mb

En este ejemplo:

- El ID de tag es utilizado por la utilería partición para identificar el nombre de los file system estándar, estos tags no afectan el acceso a la partición.
- La partición puede tener permisos de write/mountable (wm), write/unmountable (wu), read/mountable (rm), y read-only/unmountable (ru). Estas banderas (flags) no afectan el acceso a la partición.
- El inicio de los cilindros es la distancia (offset) desde el comienzo del disco.
- El tamaño de la partición puede ser expresado en bloques, cilindros o Mbytes.

6.- Introducir los nuevos valores.

En este ejemplo, los valores por omisión para el tag de identificación, las banderas de permisos de la partición y el inicio de los cilindros son aceptados, pero el tamaño de la partición es cambiado a 400Mb.

Para aceptar los valores se tiene que presionar la tecla Enter

Enter partition id tag[root]: ?

Expecting one of the following: (abbreviations ok):

Unassigned	boot	root	swap
usr	backup	stand	var
home	alternates		

Enter partition id tag[root]:

Este también trabaja con la bandera de permisos de la partición.

7.- Escribir print.

partition> **print**

Current partition table (unnamed):

Total disk cylinders available: 2036 + 2 (reserved cylinders)

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 812	400.15MB	(813/0/0)
1	swap	wu	1831 - 1983	75.30MB	(153/0/0)
2	backup	wm	0 - 2035	1002.09MB	(2036/0/0)
3	unassigned	wm	0	0	(0/0/0)
4	unassigned	wm	0	0	(0/0/0)
5	unassigned	wm	0	0	(0/0/0)
6	unassigned	wm	0	0	(0/0/0)
7	home	wm	1984 - 2035	25.10MB	(51/0/0)

En el paso anterior se escribe la instrucción print para desplegar la tabla de partición actual y poder ver el cambio hecho en el tamaño de la partición 0.

El cambio en el tamaño de la partición 0 provocó un hueco entre ésta y la partición 1 pudiendo observarse en la columna de cilindros, para compensar este espacio podría crearse otra partición en ese lugar.

8.- Escribir 1 para cambiar a la partición de swap, este paso es opcional si se desea reagrupar las particiones debido a la reducción de la partición 0.

partition> **1**

Part	Tag	Flag	Cylinders	Size	Blocks
1	swap	wu	1831 - 1983	75.30MB	(153/0/0)

Enter partition id tag[swap]:

Enter partition permission flags[wu]:

Enter new starting cyl[1831]: **813**

Enter partition size[154224b, 153c, 75.30mb]: **60mb**

En el paso anterior se le modificó el tamaño y el comienzo de los cilindros al área de swap dándole el inicio en el bloque 813 con un espacio de 60mb después de la partición 0.

Para ver el cambio efectuado es necesario introducir la instrucción print para que despliegue el estado actual de la tabla de partición.

partition> print

Current partition table (unnamed):

Total disk cylinders available: 2036 + 2 (reserved cylinders)

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 812	400.15MB	(813/0/0)
1	swap	wu	813 - 934	60.05MB	(122/0/0)
2	backup	wm	0 - 2035	1002.09MB	(2036/0/0)
3	unassigned	wm	0	0	(0/0/0)
4	unassigned	wm	0	0	(0/0/0)
5	unassigned	wm	0	0	(0/0/0)
6	unassigned	wm	0	0	(0/0/0)
7	home	wm	1984 - 2035	25.10MB	(51/0/0)

En la tabla anterior puede ser observado el cambio hecho al área de swap, pero, esto ha ocasionado que ocurra el mismo efecto que cuando se modificó la partición 0, ahora ha quedado un hueco entre el área de swap y la partición 7, para compensarlo se ajustará también esta partición.

Este paso es opcional si se desea agrupar de nuevo los cilindros.

9.- Escribir 7 para cambiar a la partición de home.

partition> 7

Part	Tag	Flag	Cylinders	Size	Blocks
7	home	wm	1984 - 2034	25.10MB	(51/0/0)

Enter partition id tag[home]:

Enter partition permission flags[wu]:

Enter new starting cyl[1984]: 935

Enter partition size[154224b, 153c, 75.30mb]: \$

partition> print

Current partition table (unnamed):

Total disk cylinders available: 2036 + 2 (reserved cylinders)

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	0 - 812	400.15MB	(813/0/0)
1	swap	wu	813 - 934	60.05MB	(122/0/0)
2	backup	wm	0 - 2035	1002.09MB	(2036/0/0)
3	unassigned	wm	0	0	(0/0/0)
4	unassigned	wm	0	0	(0/0/0)
5	unassigned	wm	0	0	(0/0/0)
6	unassigned	wm	0	0	(0/0/0)
7	home	wm	935 - 2035	541.89MB	(1101/0/0)

partition> label

Ready to label disk, continue? Yes

En este caso puede ser tomado el límite final de la última partición con el límite final del disco. Introduciendo un signo de \$ coloca la partición hasta el límite final del disco.

Es importante tomar en cuenta que para evitar que alguna partición se encime con otra se debe de empezar tomando en cuenta la importancia de las particiones que serán creadas y empezar por el cilindro 0 en un nuevo disco.

10.- Introducir la orden print para desplegar la tabla de partición completa.

11.- Cuando se está satisfecho de las particiones se debe etiquetar el disco.

El archivo format.dat

Todos los discos formateados tienen una tabla de partición como parte de su etiqueta (Fig. 3.53). Un conjunto de tablas de partición predefinidas son almacenadas en el archivo /etc/format.dat que es leída por omisión cuando la utilería format es iniciada.

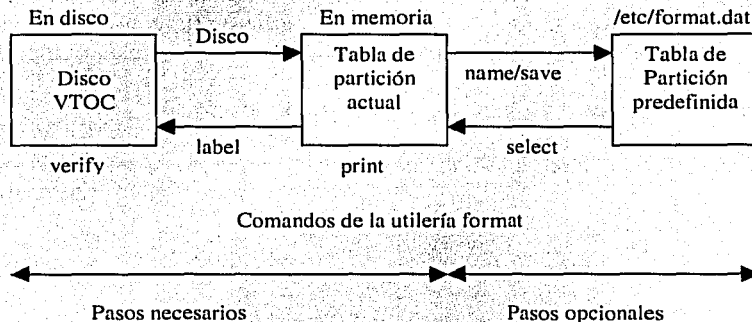


Fig. 3.53 Almacenamiento de la tabla de partición.

Cuando se selecciona un disco con formato, la actual tabla de partición almacenada en la etiqueta del disco es leída en memoria y mostrada como la actual tabla de partición.

Se puede seleccionar una tabla definida desde el archivo /etc/format.dat usando el comando `select` de la utilería format. Esta etiqueta debe ser diferente de la etiqueta actual del disco.

Una vez que se haya modificado la tabla de partición se puede almacenar en el disco o escribirla en el archivo /etc/format.dat.

El almacenar una tabla de partición en el archivo /etc/format.dat es opcional. Esto permite reutilizar una tabla de partición personalizada con la utilería format, para ello se deben de realizar los pasos:

1.- Nombra la tabla actual. Los nombres frecuentemente utilizan información del fabricante del disco.

```
partition> name
Enter table name (remember quotes): "Prod.SUN1.05"
```

2.- Salirse de la sección de partición de la utilería format.

```
partition> quit
```

3.- Salvar el disco nuevo y la definición de la partición. Al salvar ésta, es necesario obtener la ruta total donde estará colocado el archivo en el cual se salva la tabla de partición.

```
Format> save
Saving new partition definition
Enter file name["./format.dat"]: /etc/format.dat
```

Para recuperar una tabla de partición que fue previamente grabada se deben de seguir los siguientes puntos:

1.- Escribir en el prompt format la instrucción partition.

```
format> partition
```

2.- Seleccionar la tabla de partición.

```
partition> select
0. SUN1.05
1. original
2. original
3. prod.SUN1.05
```

```
Specify table (enter its number) [0] : 3
```

3.- Se debe etiquetar el disco.

```
partition> label
Are you sure? Yes
```

4.- Salirse del menú de partición.

```
partition> quit
```

5.- Salirse de la utilería format.

```
format> quit
```

Siguiendo los pasos anteriores es posible tomar una tabla de partición previamente definida que será de utilidad para cambiar la geometría de las particiones del equipo.

3.22 Respaldos

Respaldar archivos permite hacer copias de éstos, usualmente sobre medios removibles para respaldarlos en caso de pérdida o daños. Los respaldos a cintas son convenientes para restaurar archivos que son accidentalmente borrados, pero también son esenciales en caso de problemas serios a causa de fallos de hardware u otros desastres.

El respaldo de archivos es una de las funciones más importantes para el administrador del sistema. Estos pueden ser planeados para realizarse periódicamente por existir varias razones:

- Eliminación de archivos accidentalmente: Es posible que algunos usuarios o el superusuario remueva accidentalmente archivos importantes, esto puede ocasionar serias consecuencias.
- Fallas externas del sistema: Inconsistencias en la energía eléctrica pueden dañar file system o perder archivos.
- Fallas internas en el sistema: Los manejadores de disco contienen partes movibles, eventualmente estas partes pueden fallar, y entonces es necesario un respaldo para recuperar la información que contenía ese disco. Estas fallas casi nunca se presentan.
- Seguridad: Es necesario tener copias de los archivos más importantes para preveer cualquier violación por alguna persona ajena al sistema que intente realizar algún daño o que haga modificaciones al mismo sin estar autorizado.
- Factores externos: Es poco probable, pero no por ello hay que dejar de tomar en cuenta, ocurran evento difíciles de pronosticar, como por ejemplo un terremoto, ésto puede ocasionar daños al sistema o incluso dejarlo inaccesible por algún tiempo y será necesario tener algún respaldo del equipo.

Por las razones mencionadas anteriormente y por muchas más es necesario realizar respaldos del sistema periódicamente ya que éstos pueden ser de gran utilidad cuando se lleguen a presentar daños en el sistema que los ameriten o simplemente para recuperar archivos que se deseen tener de nuevo.

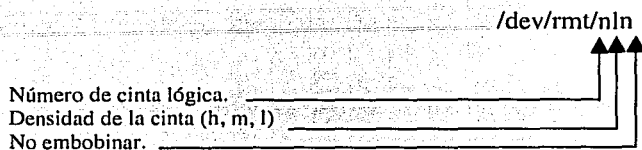
Los dispositivos de respaldo son generalmente unidades de cintas magnéticas. En la figura 3.54 se identifican cinco diferentes tipos de cintas que son disponibles pero, estas capacidades mostradas pueden ser mejoradas por lo que es necesario leer la documentación que viene con el equipo para determinar cual es la más apropiada.

Cintas	Capacidades
Cinta de ½ pulgada de carrete a carrete.	120 MB
Cartucho de ¼ de pulgada.	150MB a 8 Gbytes.
Cartucho de 8mm.	Hasta 40 Gbytes.
Cartucho de 4mm.	Hasta 24 GB. con variable incrementada usando compresión de datos
Cinta de línea digital (DLT).	Hasta 70 GB con variable incrementada usando compresión de datos

Fig. 3.54 Clases de cintas magnéticas.

Las unidades de cinta que soportan compresión de datos tienen hardware interno que ejecuta la compresión. La cinta de compresión no es tan eficientemente espaciosa como si se usara el comando de Solaris compress, éste es mucho más rápido y no requiere administración extra.

Todos los dispositivos de cinta son representados por un nombre lógico de dispositivo. El nombre de dispositivo lógico de cinta utiliza el siguiente formato:



Por ejemplo: /dev/rmt/0 → /devices/sbus@1,f8000000/esp@0,800000/st@4,0 :

Los nombres de dispositivos de cinta son numerados desde 0, independientemente del tipo y tienen varios parámetros diferentes:

Densidad de la cinta. Existen cinco valores que pueden ser dados, h (high), m (medium), l (low), c (compressed), o u (ultra). La densidad varía dependiendo del controlador de la cinta, por ello es necesario verificar la documentación para especificar la correcta. Si no es especificada, por omisión se toma el valor high.

No embobinar. Incluyendo la letra "n" al final del nombre de dispositivo, la cinta no es embobinada cuando la operación actual es completada.

Dentro de los diferentes tipos de respaldo existen :

Respalos totales (Full dumps): Respalos de un file system entero.

Respalos incrementales (Incremental dumps): Respaldo de los archivos que han sido cargados desde la última vez que se respaldó.

Respalos hechos en varios volúmenes (Multivolume dumps): Requieren más de una cinta o volumen para colocar los datos.

El comando ufsdump

El comando ufsdump es usado para respaldar file system, el respaldo puede ser total, incremental o de archivos y directorios individuales.

El formato del comando es:

ufsdump opciones [argumentos] archivos a respaldar

Existen varias opciones que pueden ser utilizadas con el comando para adquirir propiedades extras:

0-9 Especifica la opción del nivel de respaldo. El nivel 0 es el nivel más bajo, para un respaldo total y los valores 1-9 son usados para respaldos incrementales.

u Registra el respaldo actual en el archivo `/etc/dumpdates` con el día y el nivel en que se realizó, en el archivo `/etc/dumpdates` se encuentran los registros de los respaldos realizados.

f Especifica el dispositivo donde los archivos son escritos. Esto requiere un argumento que es el nombre de dispositivo.

v Verifica los datos que se colocaron en el respaldo contra los datos del file system.

Archivos a respaldar El nombre del archivo a respaldar puede ser especificado como un file system crudo o cocinado por el nombre de dispositivo (`/dev/(r)dsk/c0t0d0s0`), el nombre del file system (`/export/home`), o por el nombre de archivo o directorio a respaldar (`/etc/passwd`).

El comando `ufsdump` tiene 10 niveles de respaldo. Los niveles 1 al 9 son respaldos incrementales, éstos respaldan los archivos que cambiaron de tamaño o que fueron creados a partir del último respaldo de nivel más bajo. Si no existen respaldos de bajo nivel, será ejecutado un respaldo total, un respaldo a nivel cero 0 siempre es un respaldo total.

Los respaldos incrementales dependen de la información contenida en el archivo `/etc/dumpdates`. Desde este recurso, el comando `ufsdump` compara el día del último respaldo incremental con el día de la última modificación en los archivos individuales. Esto determina si el archivo requiere ser respaldado.

La opción `u` del comando `ufsdump` actualiza el archivo `/etc/dumpdates`. Solo los niveles de respaldo del file system son listados, no los directorios o archivos, por ejemplo:

```
# more /etc/dumpdates
/dev/rdsk/c0t2d0s6 0 Tue Dec 8 11:12:30 2000
/dev/rdsk/c0t2d0s0 3 Thu Dec 10 17:19:39 2000
/dev/rdsk/c0t2d0s4 0 Thu Dec 10 16:42:15 2000
/dev/rdsk/c0t2d0s3 0 Thu Dec 10 17:18:36 2000
/dev/rdsk/c0t2d0s6 0 Thu Dec 10 11:12:27 2000
```

El respaldo a nivel 0 es el único nivel que tiene un significado absoluto, todos los otros niveles son relacionados a otro respaldo.

Si solo se ejecutan respaldos a nivel 0 la opción `u` no es necesaria, pero si regularmente se ejecutan respaldos incrementales, el archivo `/etc/dumpdates` no debe ser manualmente modificado o borrado.

Como una sugerencia, los respaldos totales o de nivel 0 son ejecutados una vez al mes. Los respaldos de nivel 2 son realizados al final de cada semana grabando los archivos que tienen cambios a partir del último respaldo de nivel 0 hecho al inicio de mes.

Durante la semana los respaldos de nivel 3, 4, 5 y 6 pueden ser realizados.

Para realizar un respaldo total de un file system se recomienda realizar los siguientes pasos.

1.- Verificar la actividad en el sistema, ver si el sistema se encuentra ocupado por un gran número de usuarios, los respaldos deben realizarse cuando la actividad del sistema es baja, de preferencia cuando no hay usuarios en el sistema.

2.- Notificar a los usuarios acerca de la inactividad del sistema por la ejecución del respaldo y correr el sistema en modo S.

El comando `ufsdump` realiza dos pasos a los datos. El primero junta la información de los inodos, el segundo paso liga el bloque de datos. Si el file system se encuentra en actividad durante el respaldo, los archivos y directorios pueden ser modificados durante el primero y segundo paso. Esto puede ocasionar inconsistencias en algunos archivos.¹⁸

Si por alguna razón no es posible dar de baja el equipo se puede desmontar solo el file system del que se realizará el respaldo.

3.- Verificar el file system con el programa `fsck`, pero solo si ha sido desmontado.

4.- Realizar el respaldo correspondiente. A continuación se mostrará un ejemplo de un respaldo a nivel 0 del file system `/export/home` usando un cartucho de cinta.

```
# ufsdump 0uf /dev/rmt/0 /export/home
DUMP: Writing 32 Kilobyte records
DUMP: Date of this level 0 dump: Mon 30 Nov 2000 03:05:15 PM MST
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdsk/c0t3d0s7 (suned73:/export/home) to /dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 125206 blocks (61.14 MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: Tape rewinding
DUMP: 125182 blocks (61.12 MB) on 1 volume at 747 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 0 dump on Mon 30 Nov 2000 03:05:15 PM MST
#
```

Cuando el dispositivo de la unidad de cinta utiliza la dirección por omisión (`dev/rmt/0`), no es necesario especificar la unidad con la opción `f`. La siguiente cadena realiza la misma función que la introducida para realizar el respaldo del ejemplo anterior:

```
# ufsdump 0u /export/home
```

¹⁸ Cfr. Sun Educational Services. *Solaris 7 System Administration I*. Cap. 17

El comando `ufsdump` y `ufsrestore` (`ufsrestore` será visto más adelante) pueden ser usados para realizar respaldos o restaurar la información desde un dispositivo de cinta remoto.

Para realizar un respaldo o una restauración de archivos remota se debe de tomar en consideración dos puntos importantes.

- Tener privilegios de root para poder entrar al sistema remoto sin la solicitud de `passwd` (estar registrado en el archivo `/etc/hosts`).
- Especificar en la cadena del comando `ufsdump` o `ufsrestore` el servidor y la ruta de la unidad de cinta (*servidor:unidad_de_cinta*).

El siguiente ejemplo mostrará cómo se realiza un respaldo a nivel 0 del file system `/export/home` en un servidor remoto.

```
# ufsdump 0uf cmr:/dev/rmt/0 /export/home
```

```
DUMP: Writing 32 Kilobyte records
DUMP: Date of this level 0 dump: Mon 30 Nov 2000 03:05:15 PM MST
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdisk/c0t3d0s7 (suned73:/export/home) to cmr:/dev/rmt/0.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 125206 blocks (61.14 MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: Tape rewinding
DUMP: 125182 blocks (61.12 MB) on 1 volume at 747 KB/sec
DUMP: DUMP IS DONE
DUMP: Level 0 dump on Mon 30 Nov 2000 03:05:15 PM MST
#
```

Hasta ahora se ha visto cómo realizar respaldos de información con el comando `ufsdump`, a continuación se dará la explicación necesaria para restaurarla.

El comando `ufsrestore`

El comando `ufsrestore` extrae los archivos desde un respaldo creado con el comando `ufsdump`, el formato del comando es:

```
ufsrestore  opciones  [argumentos]  nombre de archivos
```

Al igual que la mayoría de las utilerías del sistema operativo el comando `ufsrestore` tiene opciones que proporcionan características adicionales para un mejor uso de este.

Opciones:

- i La opción i permite la restauración de los archivos de una forma interactiva, es decir, el sistema preguntará si se desean realizar los pasos que se mostrarán durante la ejecución de la restauración.
- r La opción r permite restaurar el respaldo total.
- t La opción t permite listar la tabla de contenido del respaldo, es decir, solo los nombres de los archivos que se encuentran dentro del respaldo.
- x Esta opción permite restaurar solo los archivos que se coloquen en la línea de comandos (solo los archivos especificados).
- f La opción f permite especificar la ruta del dispositivo donde fue respaldada la información para poderlo restaurar desde esa misma ruta (/dev/rmt/0).
- v La opción v permite desplegar la ruta de los nombres de archivo como serán restaurados.

Es importante tomar en cuenta que el tamaño de los archivos en una cinta de respaldo con el comando ufsdump los coloca de acuerdo a un nombre de ruta relativa, cuando se realiza una restauración, el árbol de directorios y archivos es colocado en el directorio actual, por esto es recomendable que al momento de bajar un respaldo se haga sobre un directorio temporal y de ahí sean movidos a la localidad correcta dentro del file system.

En el siguiente ejemplo se ejecutará la restauración de un archivo dentro de un directorio temporal y será movido a su localidad original, los pasos son los siguientes:

1. Cargar la cinta dentro de la unidad.
2. Conectarse al sistema como superusuario.
3. Colocarse dentro de un directorio temporal para evitar que se sobrescriban archivos (/var/tmp).
4. Desplegar el contenido de la cinta para verificar los archivos que contiene e identificar la ruta correcta del archivo que será restaurado. En este ejemplo, el archivo por omisión /dev/rmt/0 es dado porque la opción f requiere del argumento.

```
# ufsrestore tvf /dev/rmt/0
2
3 ./lost+found
5120 ./export
10240 ./home
25600 ./etc
30720 ./etc/passwd
```

5. Una vez que se han verificado los archivos que contiene y se ha seleccionado el correcto debe ser extraído el archivo a restaurar.


```
# ufsrestore xvf /dev/rmt/0 /etc/passwd
you have not read any volumes yet.
Unless you know which volume your file(s) are on you
should start with the last volume and work towards
the first.
Specify next volume #: 1
Set directory mode, owner, and times.
set owner/mode for '..?' [yn] n
```

En el ejemplo anterior pregunta el sistema si se desea cambiar los permisos de los archivos, para ello basta con escribir una "y" si se desea hacer, o una "n" si no se desea hacerlo.

6. Se debe colocar el número de volumen que contiene el archivo que se desea extraer. El número de volumen es el 1 ya que el respaldo total solo se realizó en una cinta.
7. Verificar que el archivo se posicionó en el lugar esperado, que es el correcto y moverlo a la localidad correcta.

La restauración anterior puede decirse que fue hecha de forma automática, el comando ufsrestore permite realizar restauraciones de forma interactiva, para ello se mostrará un ejemplo de cómo recuperar un archivo con este método.

1.- Posicionarse en un directorio temporal, esto es para evitar sobre escribir el archivo y para poder verificarlo antes de moverlo al lugar correcto.

2.- Iniciar el comando ufsrestore con la opción interactiva.

```
# ufsrestore ivf /dev/rmt/0
verify volume and initialise maps
media block size is 64
Dump date: Mon Nov 30 15:30:23 2000
Dumped from: the epoch
Level 0 dump of / on rcola:/dev/dsk/c0t3d0s0
Label: none
Extract directories from tape
Initialise symbol table.
```

3.- Desplegar el contenido de la estructura del directorio en el volumen del respaldo.

```
ufsrestore > ls
2          */          39         devices/    30847      net/
2          */          5122      etc/        15360      opt/
161       .Xauthority  5120      export/    25611      proc/
160       .Xdefaults  10240     home/      15381     /sbin/
159       .rhosts    40        kadb       35863      tmp/
3085     .wastebasket/ 25608     kernel/    30848      tmp_mnt/
3        bin        35        lib        20480      usr/
3087     cdrom/    3         lost+found/ 25600     var/
25610    dev/        20503     mnt/
```

Es posible cambiarse de directorio dentro del volumen de respaldo.

```
ufsrestore > cd etc/inet
ufsrestore > ls
```

4.- Una vez que se está posicionado en el directorio que contiene los archivos hay que adicionarlos a la lista de archivos que van a ser extraídos.

```
ufsrestore > add inetd.conf hosts
```

Los archivos son marcados con un asterisco (*). Si se marca un directorio para extraerlo, todo su contenido es marcado para su extracción.

En este ejemplo, dos archivos fueron marcados, por lo tanto, la salida desde el comando ls incluye todos los archivos del directorio /etc/inet con asterisco al frente *hosts y *inetd.conf.

Para borrar un archivo de la lista de los que van a ser extraídos, se usa el comando delete:

```
ufsrestore > delete inetd.conf
```

5.- Una vez que ya se tienen los archivos seleccionados deben ser extraídos.

```
ufsrestore > extract
Extract requested files
you have not read any volumes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
extract file ./etc/inet/hosts
Add links
Set directory mode, owner, and times.
set owner/mode for '??' [yn] n
```

6.- Ya que los archivos han sido extraídos, se debe salir de la restauración interactiva.

```
ufsrestore > quit
```

La restauración de archivos o directorios por medio del comando ufsrestore contiene varias opciones más que pueden ser utilizadas para casos específicos, solo hay que desplegar la ayuda sobre el comando.

El comando tar

Otro comando muy utilizado para respaldar archivos y directorios es el comando tar (tape archive) que permite respaldar uno o múltiples archivos en un directorio jerárquico. El formato del comando es:

```
tar      opciones      [argumentos]      nombre de archivos
```

Existen varias opciones del comando tar que permiten realizar funciones específicas:

- c Esta opción crea un respaldo de los archivos especificados en la línea de comandos.
- t La opción t permite desplegar el contenido del archivo de respaldo.
- x Esta opción permite extraer los archivos especificados desde el archivo de respaldo, si no se especifican los nombres, la información contenida en el respaldo total es extraída.
- v Esta opción imprime los nombres de los archivos que son restaurados.
- f Esta opción permite especificar la ruta del dispositivo donde será respaldada la información.

El siguiente ejemplo muestra cómo utilizar el comando tar para respaldar un directorio en el dispositivo manejador de cinta por omisión.

```
# tar cv ./salhade/
a ./salhade/ 0 tape blocks
a ./salhade/java/ 0 tape blocks
a ./salhade/java/test.class 2 tape blocks
a ./salhade/java/research.class 2 tape blocks
a ./salhade/solaris/ 0 tape blocks
```

Para verificar el respaldo realizado se utiliza el comando tar con la opción t.

```
# tar t
a ./salhade/
a ./salhade/java/
a ./salhade/java/test.class
a ./salhade/java/research.class
a ./salhade/solaris/
```

El comando tar es de gran utilidad cuando se desea respaldar archivos o directorios pequeños sin tantas opciones.

El comando *cpio*

La orden *cpio* fue creada para sustituir a *tar* (*tar* es una orden primitiva del sistema UNIX creada para colocar archivos en cinta). La principal ventaja de *cpio* con respecto a *tar* es su flexibilidad. *cpio* acepta su entrada desde la entrada estándar y dirige su salida a la salida estándar.

Así por ejemplo, se puede dar a *cpio* una lista de archivos a guardar mediante el listado de los contenidos de un directorio (*ls*) o la impresión de un archivo que contenga la lista (*cat*). Puede entonces dirigir la salida de *cpio*, que es un archivo de *cpio*, a un disco flexible, una cinta u otro medio, incluyendo un disco flexible.

El comando *cpio* funciona en tres modos: modo de salida (*cpio -o*), modo de entrada (*cpio -i*) y modo de paso (*cpio -p*).

* Modo de salida (copia de seguridad). A *cpio -o* se le proporciona una lista de archivos y un destino. *cpio* empaqueta entonces esos archivos en un único archivo de *cpio* (con información de cabecera) y copia ese archivo en el destino.

* Modo de entrada (restauración). A *cpio -i* se le proporciona un archivo de *cpio*. *cpio* divide entonces el archivo de nuevo en archivos y directorios separados y los repone en el sistema. (Pueden restaurarse únicamente archivos seleccionados.)

* Modo de paso. *cpio -p* funciona como el modo de salida, excepto que en vez de copiar los datos a un archivo, cada archivo se copia (o vincula) individualmente a otro directorio en el árbol de sistema de archivos. Con esta característica se pueden hacer copias de seguridad en otro disco o en un sistema remoto montado en el sistema.

A continuación se muestran unos cuantos ejemplos de *cpio*.

El siguiente ejemplo usa la orden *find* para recolectar los archivos a copiar, proporciona esa lista de archivos a *cpio* y los copia en un disquete cargado en la unidad 1.

```
# find . -depth -print | cpio -ocv > /dev/diskette1
```

Para restaurar archivos desde un disco flexible se puede pasar al directorio sobre el que desea restaurarlos y utilizar la orden *cpio -i*.

```
# cpio -ivcd < /dev/diskette1
```

El siguiente ejemplo utiliza *cpio -p* para transferir copias de archivos desde un punto en la estructura de directorios hasta otro punto. En este caso todos los archivos por debajo del directorio propio de un usuario serán copiados a un sistema de archivos remoto.

```
# find . -depth -print | cpio -pmvd /mnt/backup
```

El comando mt

Cuando se trabaja con respaldos a cinta existe la necesidad de manipular la cinta, el comando mt y sus opciones permiten tener control de esta, el formato del comando es:

```
mt [-f nombre del dispositivo de cinta] comando [count]
```

La opción -f es usada para especificar el nombre del archivo de dispositivo de cinta, si la opción es omitida el valor de la variable de ambiente TAPE es usada para permitir la manipulación del dispositivo de cinta. count dentro de la línea de comando de la instrucción mt permite especificar el número de archivos, esto puede utilizarse en algunos comandos de mt.

Los comandos de la instrucción mt son:

status	Despliega información acerca del estado actual del manejador de cinta.
rewind	Regresa la cinta al inicio.
retension	Este comando permite que la cinta se adelante y regrese al inicio para eliminar la tensión de la cinta.
erase	Borra el contenido de la cinta entera.
fsf	Brinca hacia delante [count] un número de archivos en la cinta.
bsf	Brinca hacia atrás [count] un número de archivos en la cinta.
nbsf	Brinca hacia atrás [count] un número de archivos en la cinta. La cinta es posicionada en el primer bloque del archivo.
eom	Se posiciona al final de la cinta.

Los comandos de mt mencionados anteriormente son los más utilizados para un mejor manejo de la cinta.

Pasos para adicionar un disco al sistema

Una de las labores que realiza el administrador del sistema es la de adicionar hardware al equipo, a continuación se dará el ejemplo de los pasos que deben de realizarse para conectar un nuevo disco al sistema.

- 1.- Colocar el hardware, asignándole una dirección física al dispositivo mediante un switch que contiene el dispositivo, conectar los cables del dispositivo siguiendo las instrucciones que vienen con él.

- 2.- Una vez conectado físicamente el disco es necesario reiniciar el sistema con el comando boot -r para que cree la nueva estructura de árbol tomando en cuenta el nuevo dispositivo.
- 3.- Una vez que es reconocido el disco, será necesario crearle particiones.
- 4.- Ya que se le han creado las particiones, es necesario crear los file systems.
- 5.- Si se desea que los file systems del nuevo disco sean montados automáticamente es necesario editar la tabla del archivo /etc/vfstab.
- 6.- Reiniciar el sistema para que los cambios sean tomados y verificar que no mande algún mensaje de error.

Como se puede observar la instalación de un dispositivo físico requiere una serie de pasos que agrupan conocimientos de diferentes temas de la administración del Sistema Operativo Solaris, el ejemplo anterior sirve de muestra para darnos cuenta de que todos los temas mencionados para la implantación del disco fueron comentados en este trabajo de tesis y de que esta recopilación de información puede dejar al lector capacitado para realizar las funciones principales del administrador, cumpliéndose así el objetivo principal de este trabajo.

3.23 Algunos comandos de administración avanzada

A continuación se explicarán brevemente algunos comandos que desde mi punto de vista ya pertenecen a la administración avanzada puesto que se requiere de conocimientos más amplios para poder entender su salida. Estos comandos en algún momento pueden ser de utilidad por lo que se presentan aquí pero para un mayor desarrollo se recomienda consultar los manuales de administración avanzada o administración de redes.

El comando iostat

El comando iostat reporta la actividad del disco a la terminal. Este comando puede ser usado para medir como están dispersas las cargas en cada disco del sistema servidor.

En el siguiente ejemplo, el comando iostat mide la carga de los discos cada 5 segundos.

```
# iostat -D 5
```

La primera línea de la salida resume las estadísticas desde el primer inicio del sistema. Cada línea siguiente representa la actividad desde el intervalo específico.

El siguiente ejemplo muestra un servidor con una carga de disco mal balanceada.

iostat -D 5

sd0			sd1			sd2			sd3		
rps	wps	util	rps	wps	util	rps	wps	util	rps	wps	Util
0	0	0.0	0	0	0.2	0	0	0.2	19	0	56.5
0	1	2.6	0	0	0.0	0	0	0.0	0	17	99.2
4	0	8.0	0	0	0.0	0	0	0.0	14	0	89.3
0	2	2.3	0	0	0.0	0	0	0.0	7	17	78.0
:	:	:	:	:	:	:	:	:	:	:	:

En este ejemplo se puede comparar el campo útil para los discos representados por las columnas etiquetadas como sd0, sd1, sd2, y sd3. Nótese que el disco sd3 es pesadamente usado por la combinación de escrituras y lecturas (rps y wps). Los otros discos sd0, sd1 y sd2, son muy poco utilizados.

El siguiente ejemplo muestra un servidor con mejor balance de la carga en los discos.

iostat -D 5

sd0			sd1			sd2			sd3		
rps	wps	util	rps	wps	util	rps	wps	util	rps	wps	Util
6	8	8.0	19	0	40.2	0	17	20.2	19	0	25.5
0	1	2.6	14	0	36.80	0	14	45.0	17	0	39.2
4	0	8.0	17	0	27.0	0	22	37.0	14	0	29.3
0	2	2.3	15	0	23.0	0	8	23.3	7	1	38.0
:	:	:	:	:	:	:	:	:	:	:	:

En el ejemplo anterior puede verse cómo la carga es dispersada por todos los discos, y cada disco es dedicado casi a escribir o leer.

iostat es un comando útil para la configuración del equipo por medir la utilización y el tipo de actividad (read/write) en cada disco, el administrador del sistema puede escoger una óptima localidad para datos en una máquina que será compartida con clientes usando file systems NFS.

Por ejemplo, cuando se configura una máquina para soportar clientes adicionales, el comando iostat puede correrse para identificar los niveles de utilización del disco pensando en tener una distribución de la carga óptima en los discos, el administrador puede utilizar las estadísticas del comando para identificar la localidad ideal para el cliente root, swap, áreas de arquitectura y el mejor lugar para los directorios *home* de los usuarios.

El comando vmstat

El comando vmstat reporta estadísticas sobre procesos, memoria virtual, discos, trap y actividad del CPU.

Lo más importante, los cuellos de botella en CPU y memoria son rápidamente localizados con el comando vmstat.

```
# vmstat 5
```

Procs			memory				page				disk				faults			cpu	
r	b	w	swap	free	re	mf	pi	po	fr	de	sr	s3	s5	in	sy	cs	us	sy	Id
1	0	5	516	0	0	1	0	0	0	0	0	0	0	9	60	48	4	2	94
0	0	3	75980	0	0	13	184	236	236	0	76	1	15	187	245	127	6	14	80
1	0	4	75984	0	0	0	96	40	40	0	60	0	6	148	152	100	7	5	88
0	0	2	75984	0	0	0	120	176	176	0	53	0	4	131	84	86	6	5	89

Encolamientos

Los campos que aparecen bajo la columna de título *procs* describen el número de procesos residentes en la cola de ejecución. Los valores que aparecen en el campo etiquetado como *r* representan procesos cargados en memoria. El siguiente campo (*b*) describe el número de procesos que no pueden ser ejecutados porque están esperando operaciones de disco, terminal, o red para ser completados. Los campos que se muestran en la columna *w* son encontrados fuera de área de swap, en memoria.

Memoria Virtual

La columna *sr* indica el número de revoluciones del demonio de la página por segundo. Este demonio corre solo cuando la memoria disponible es baja. Valores continuos sobre 20 por segundo indican un déficit de memoria.

Utilización de CPU

La columna del CPU describe la utilización basada en usuarios y llamadas de sistema. Continuos valores bajos (menos de 10) que aparecen en el campo *id* de la columna *cpu* indican que el CPU está muy ocupado.

Muchos procesos corriendo en el área de swap (son indicados en el campo *procs*), junto con pequeños procesos que corren en memoria (son indicados en el campo *sr*), indican una memoria física escasa.

En el siguiente ejemplo, la memoria virtual es usada pesadamente y las entradas y salidas del disco pueden volverse un cuello de botella.

vmstat 5

Procs		memory			page				disk				faults			cpu			
r	b	w	swap	Free	re	Mf	pi	po	fr	de	sr	s3	s5	in	sy	cs	us	sy	Id
1	0	5	516	0	0	1	0	0	0	0	0	0	0	9	60	48	4	2	94
0	0	3	75980	0	0	13	184	236	236	0	76	1	15	187	245	127	6	14	80
1	0	4	75984	0	0	0	96	40	40	0	60	0	6	148	152	100	7	5	88
0	0	2	75984	0	0	0	120	176	176	0	53	0	4	131	84	86	6	5	89

El comando swap

El comando swap es usado para desplegar el espacio usado de *swap*, y para adicionar o remover áreas de swap en disco.

El siguiente ejemplo usa la opción *-l* para listar el número total de sectores de disco (blocks) y sectores sin usar (free) en cada dispositivo de swap.

swap -l

```
swapfile          dev          swaplo        blocks        free
/dev/dsk/c0t3d0s1 32,25         8             187915        127088
```

Creación y adición de áreas de swap

Para adicionar áreas de swap puede realizarse por medio de archivos regulares (creados con el comando *mkfile*) o por partición de disco. El siguiente ejemplo muestra como crear un área de swap de 10 Mb. Con el comando *mkfile*.

mkfile 10m /exp/swap

El siguiente paso después de que se creó el área de swap es adicionarla usando la opción *-a* del comando swap. El siguiente ejemplo muestra cómo adicionarla y comprobar su adición.

swap -a /exp/swap

#

swap -l

```
swapfile          dev          swaplo        blocks        free
/dev/dsk/c0t3d0s1 32,25         8             187912        126920
/exp/swap         -             8             20472         20472
```

Eliminación de áreas de swap.

Usando la opción `-d` del comando `swap` se pueden remover áreas de swap. En el siguiente ejemplo, el área de swap previamente adicionada será removida, y el ajuste del área será verificado.

```
# swap -d /exp/swap
```

```
#
# swap -l
```

swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t3d0s1	32,25	8	187912	126920

Si el área de swap contiene datos que están siendo usados por procesos, el comando `swap` les dará una nueva ubicación antes de borrar el área.

El comando netstat

El comando `netstat` despliega información relacionada con la red, incluyendo estadísticas para cada interfaz de red.

Colisiones

En el siguiente ejemplo, un intervalo de 10 segundos es usado para medir el rango de colisiones en los paquetes.

```
# netstat -i 10
```

input		le0		output		input		(total)		output
packets	errs	packets	errs	colls	packets	errs	packets	errs	colls	
1929138	0	1590861	3	73176	1946182	0	1607905	3	73176	
16	0	1	0	0	16	0	1	0	0	
3	0	3	0	0	3	0	3	0	0	
.	
.	

El rango de colisión para una simple máquina es el número de colisiones de salida dividido por el número de paquetes enviados. El rango de colisiones para la red se obtiene:

**TESIS CON
FALLA DE ORIGEN**

Obtener las estadísticas de red corriendo este comando sobre todas las máquinas activas, sacar la suma de todas las colisiones de los equipos y dividirlo entre el número total de los paquetes enviados.

El rango de colisión es iniciado con las colisiones de los paquetes enviados dividido por el total de los paquetes de salida multiplicado por 100. Usando los valores del ejemplo anterior queda:

$$73176 / 1590861 * 100 = 4.6\% \text{ rango de colisión.}$$

Un rango de colisión más alto del 5% es considerado como la primera indicación de carga en la red.

Colisiones más altas del 10% significan una gran carga en la red que debería considerarse para pensar en fragmentar la red.

El comando snoop

El comando snoop captura y despliega el contenido de los paquetes en una red. snoop puede ser usado para localizar el origen y destino de las anomalías de red.

La salida de este comando puede rápidamente saturar el espacio disponible en disco con la información capturada. Algunas sugerencias de cómo utilizar el comando para la captura de paquetes son:

- Capturar un archivo binario dirigiendo la salida a un archivo.

```
# snoop -o nombre de archivo
```

CTRL-c

Para desplegar el contenido del archivo binario

```
# snoop -i nombre de archivo
```

- Captura solo información de interés como el contenido de la cabecera de los paquetes localizados en los primeros 120 bytes.

```
# snoop -s 120
```

- Captura un número específico de paquetes.

```
# snoop -c 1000
```

- Usa filtros para capturar un tipo específico de actividad de red.

```
# snoop broadcast
```

El comando mpstat

El comando mpstat reporta estadísticas de información sobre el uso del procesador. Cada fila en la tabla representa la actividad del procesador. La primera tabla resume toda la actividad al encendido del equipo, cada tabla siguiente reporta la actividad de los intervalos siguientes. Por ejemplo:

```
# mpstat 5
```

CPU	minf	mjf	xcal	intr	ithr	csw	icsw	migr	sntx	srw	syscl	usr	sys	wt	idl
20	40	0	473	261	237	144	30	19	38	0	546	76	7	2	15
22	38	0	535	69	46	165	30	18	10	0	524	77	7	2	15
24	41	0	478	257	42	180	29	19	11	0	510	74	7	3	16
26	42	0	498	26	0	194	31	20	11	0	489	74	7	3	16

En el ejemplo anterior existen varias filas de las cuales solo se mencionaran:

- usr: En este campo se puede ver el porcentaje de procesador utilizado por los usuarios.
- sys: Porcentaje de procesador usado por el sistema.
- wt: Porcentaje de procesador en espera.
- idl: Porcentaje de procesador disponible.

El comando sar

El comando sar reporta la actividad del sistema. sar reporta la utilización de los recursos en forma interactiva o a través de procesos batch.

El siguiente ejemplo interactivo mide la utilización del CPU y toma dos muestras separadas por 10 segundos.

```
# sar 10 2
```

```
Sun OS seasun2 5.2 Generic sun4c 05/02/01
```

	%usr	%sys	%wio	%idle
01:53:15				
01:53:24	0	0	0	100
01:53:34	0	1	2	97
Average	0	0	1	99

Reporte Batch

Si la opción `-o` es utilizada, los ejemplos son guardados en un archivo en formato binario.

```
# sar -o /tmp/sar.arch 10 2
```

La información colectada en el archivo puede leerse usando la opción `-f`. Por ejemplo:

```
# sar -f /tmp/sar.arch
```

```
Sun OS  season2  5.2  Generic  sun4c  05/02/01
```

01:53:15	%usr	%sys	%wio	%idle
01:53:24	0	0	0	100
01:53:34	0	1	2	97
Average	0	0	1	99

Utilización de disco

Usando la opción `-d` del comando `sar` pueden buscarse cargas disperejas en los discos. En el siguiente ejemplo puede verse el disco `sd3`, está más ocupado que los otros dispositivos, esto es indicado en la columna `%busy`.

```
# sar -d 10 2
```

```
Sun OS  season2  5.2  Generic  sun4c  05/02/01
```

02:03:05	device	%busy	avque	r+w/s	blks/s	await	avserv
02:03:15	fd0	0	0.0	0	0	0.0	0.0
	sd2	0	0.0	0	0	0.0	0.0
	sd3	67	0.7	9	2206	0.0	75.5
02:03:25	fd0	0	0.0	0	0	0.0	0.0
	sd2	2	0.2	1	22	88.5	46.1
	sd3	69	0.7	9	2182	0.0	79.7
Average	fd0	0	0.0	0	0	0.0	0.0
	sd2	1	0.1	1	11	88.5	46.1
	sd3	68	0.7	9	2194	0.0	77.6

Memoria virtual

Usando la opción `-g` del comando `sar` se puede monitorear el uso de la memoria virtual. La columna `pgscan/s` indica el número de páginas examinadas por segundo por el demonio `pageout`. Este demonio corre solo cuando la memoria disponible es baja. Continuos valores sobre 20 por segundo indican memoria deficiente.

```
# sar -g 10 2
Sun OS seasun2 5.2 Generic sun4c 05/02/01
```

	pgout/s	ppgout/s	pgfree/s	pgscan/s	%ufs_ipf
02:22:34	1.20	12.08	21.06	23.85	7.06
02:22:54	1.10	9.00	19.30	21.30	10.81
Average	1.15	10.54	20.18	22.58	7.35

Encolamientos

Usando la opción `-q` del comando `sar` se puede buscar cuantos procesos están encolados para su ejecución. La columna `runq-sz` indica el número de procesos ligados a memoria en espera para su ejecución. Valores continuos más altos que 2 indican que el CPU del sistema está al límite

La columna `%runocc` indica cuando la cola de procesos listos para ejecutarse está ocupada. Valores pequeños son preferibles ya que indican una buena distribución del tiempo de procesamiento para cada proceso en cola

```
# sar -q 10 2
Sun OS seasun2 5.2 Generic sun4c 05/02/01
```

	Runq-sz	%runocc	Swpq-sz	%swpocc
02:35:31	12.6	100	0	
02:35:41	12.0	100	0	
02:35:51	12.0	100	0	
Average	12.3	100	0	

Memoria libre

Usando la opción `-r` del comando `sar` se puede buscar el promedio de las páginas de memoria disponible y los bloques de disco libres.

La columna `freemem` indica el promedio de páginas de memoria disponible para procesos de usuario en cada intervalo.

La columna `freeswap` muestra cuántos sectores de disco de 512-bytes son disponibles para el swap.

Continuos valores de memoria bajos `freemem` (menores del 6% en memoria instalada) indican una baja capacidad de memoria y un potencial cuello de botella.

```
# sar -r 10 2
Sun OS seasun2 5.2 Generic sun4c 05/02/01
```

	freemem	freeswap
02:55:11	2686	178006
02:55:22	2208	160269
02:55:32	2208	160269
Average	2447	169151

RESUMEN CAPÍTULO III

Solaris es el sistema operativo utilizado en los equipos SUN, es un sistema multiusuario y multitarea que presenta variados beneficios, lo que implica también ciertas responsabilidades para establecer, mantener y solucionar las dificultades que surjan durante su funcionamiento, es decir, implica una adecuada administración del sistema.

El administrador del sistema es la persona encargada de administrar los recursos del equipo a su cargo, es responsable de la instalación, configuración, mantenimiento y reparación del Sistema Operativo, pues está facultado para realizar tareas inaccesibles a los usuarios ordinarios. Entre sus funciones se encuentran: instalación y pruebas de hardware, planeación, instalación del sistema operativo, configuración, operaciones rutinarias, administración de recursos, instalación y configuración en red, realizar medidas de seguridad, análisis de fallas, respaldos, actualizaciones, y asistencia a los usuarios en sus dudas sobre el manejo del equipo.

El proceso de instalación del equipo es una de las tareas elementales del administrador del sistema. La instalación es un proceso sencillo pero delicado, pues consiste en la adecuada colocación de los componentes del equipo; consta de dos etapas: instalación del hardware e instalación del software, la primera dirigida a los componentes físicos del equipo y la segunda a los programas para su operación.

La conexión física en un equipo Sun cuenta con puertos específicos en la parte posterior, donde son conectados los diferentes dispositivos, además, este equipo dispone del puerto SCSI, que es un bus al que se le pueden conectar hasta 7 dispositivos. La conexión y encendido de los dispositivos del equipo debe seguir ciertos pasos para que funcionen adecuadamente.

Los sistemas Sun cuentan con programas de chequeo básico del hardware; el OpenBoot PROM, es un programa de solo lectura residente del sistema Sun que provee pruebas básicas al hardware para manipularlo por medio del cambio de parámetros, siendo al mismo tiempo, una interfaz de usuario. El Boot PROM contiene pruebas de diagnóstico, manejador de dispositivos, interfaz de usuario y control de cambios de parámetros; la interfaz de usuario del Open Boot es el PROM Monitor, que cuenta con diferentes comandos de inicialización del equipo, cada uno de ellos con finalidades específicas.

El sistema Solaris 7 simplifica la acción de direccionamiento para la asociación de un manejador de dispositivo con un dispositivo periférico, mediante la asignación única de controladores que permiten al administrador del sistema controlar en su totalidad dicho dispositivo por medio de una tarjeta de direcciones.

Una vez terminado el proceso de instalación física del equipo, se procede a la instalación del sistema operativo; el administrador del sistema deberá saber qué tipo de configuración desea para el equipo, pues existen diferentes configuraciones con características particulares acordes a los diferentes requerimientos.

La instalación del software está conformada por paquetes, clusters y configuración de clusters; los paquetes son grupos de software con programas interrelacionados almacenados en archivos y directorios, al momento de la instalación tales paquetes son agrupados en clusters -colecciones lógicas de paquetes-, que pueden incluir uno o más paquetes. Existen cinco diferentes formas de instalar el sistema operativo y configurar los clusters: Entire Distribution plus, Entire Distribution, Developer, End user, y Core, los tamaños de las configuraciones se basan en el tipo de máquina y el número de clientes que soportará el sistema.

Para instalar el sistema operativo se utiliza la herramienta del sistema SunInstall, que inicia este proceso con la solicitud y llenado de información para identificar el sistema. Al instalar por vez primera el sistema operativo con el SunInstall, es necesario tener la información que el programa solicitará para identificar el sistema y configurarlo en red. Para actualizar la instalación del sistema operativo se llevan a cabo algunos pasos adicionales, como acceder al sistema como superusuario, notificar a otros usuarios sobre la actualización, respaldar todo el sistema, dejar el equipo en nivel de ok., insertar el CD en la unidad y mandar la instrucción de arranque desde él. Todo el proceso de instalación del sistema operativo Solaris 7 se presenta en una secuencia de ventanas de fácil determinación.

El proceso de instalación inicial del software se presenta en una serie de ventanas informativas sobre los eventos que se desarrollarán durante él; éstas despliegan información sobre el inicio de la instalación, identificación del tipo de sistema, selección de lenguaje adicional, tipo de instalación, selección de discos de instalación, selección de manejador de discos, guardado de datos existentes, partición del disco y file systems, file system remoto, profile, reboot, notificación del estadio de instalación (ventana de progreso), root password, donde solicita la inserción del password de root (superusuario), y, finalmente, aparece una ventana que notifica que el sistema está listo.

El administrador del sistema debe conocer entonces, tanto el procedimiento de instalación física, como la instalación del software; en este sentido, se puede definir la administración del sistema como el conjunto de tareas que ejecuta el propietario de una computadora para que ésta marche adecuadamente; así, el administrador debe estar al tanto de las tareas de administración del sistema, teniendo en cuenta el tipo de sistema operativo, el tamaño del sistema en general, aplicaciones, usuarios conectados a él, programas y dispositivos compatibles, etc, además, debe depurar el sistema operativo, optimizar discos y hacer copias de seguridad para mejorar su funcionamiento.

La administración del sistema operativo en un equipo SUN se divide en básica y avanzada, aunque no existe un criterio definido para separarlas detalladamente, pues el nivel de la administración depende de la dimensión y propiedades del equipo, así como de los conocimientos del administrador y las soluciones que pueda dar a los problemas derivados del equipo.

Sin embargo, existen conceptos básicos que el administrador del sistema debe conocer para el manejo de cualquier equipo que cuente con el sistema operativo Solaris, como las medidas que se siguen para encender y apagar el equipo, los pasos de diagnóstico-verificación del Boot PROM, el archivo de arranque */etc/inittab*, etcétera.

El sistema Solaris cuenta con diferentes estados del sistema (modos de operación), que le permiten al administrador limitar la actividad del sistema cuando realice determinadas labores de administración; este sistema posee dos estados de sistema de ejecución: monousuario (sólo está montado el sistema de archivos de raíz) y multiusuario (se montan todos los sistemas de archivos). Por default, el sistema arranca en estado multiusuario; la cantidad de servicios disponibles del sistema varían según el estado de sistema en que se encuentre. Un superusuario puede utilizar el comando *init* para cambiar de un estado de sistema a otro, el comando *shutdown* se emplea para cambiar el nivel al que corre el sistema.

Por otro lado, el sistema operativo contiene archivos de administración (sistemas de archivos agrupados que contienen información muy valiosa para el equipo y los usuarios), que deben ser conocidos y manejados por el administrador a fin de operar el equipo adecuadamente; estos archivos se ubican en raíz (*/*), */home*, */usr*, y */var*.

La instalación del software de Solaris se distribuye en paquetes que son seleccionados y adicionados automáticamente desde el CD ROM mediante los paquetes de administración del sistema o la interfaz gráfica *Admintool*, estos paquetes contienen una reseña de los mismos, una descripción de la relación del sistema destino, los archivos que serán instalados y programas de pre y post instalación.

El administrador del sistema debe dominar los comandos sobre instalación de software, para ello es preciso que conozca y maneje perfectamente los comandos *pkginfo*, *pkgrm*, *pkgadd*, *pkgchk*, y */var/sadm/install/contents*, que son comandos de instalación y ubicación de paquetes en el sistema. Existen algunos paquetes que pueden ser utilizados sin estar instalados en el sistema, son los llamados paquetes *spooling*, que pueden ser copiados desde el CD ROOM y almacenados dentro del sistema sin ser instalados.

El administrador debe tener conocimiento del funcionamiento de estos paquetes y su instalación, así como de la instalación de parches. Un parche es un conjunto de archivos o directorios que reemplazan o actualizan archivos o directorios para ejecutar algún programa de software, se encarga de actualizar el entorno del sistema para que todas las funciones puedan reconocer las nuevas aplicaciones.

Una parte fundamental en la administración del sistema operativo Solaris es el conocimiento y manejo y los procesos del sistema (programas de ejecución), identificados mediante el número asignado a cada uno (PID); un proceso se crea por medio de una llamada del sistema *fork*, el proceso que hace la llamada a este sistema se le conoce como proceso padre, si se crea otro proceso después de la llamada se le denomina proceso hijo.

Para agilizar los procesos y ejecutarlos simultáneamente, el shell puede crear procesos hijos que, al morir, permiten que el padre despierte y se ejecute, o bien, funcionen ambos ejecutando múltiples procesos al mismo tiempo; el comando *ps* (estado de proceso) permite visualizar los procesos que están corriendo en el sistema, el comando *kill* se utiliza para detener algún proceso subordinado o asociado a otra terminal en ejecución.

Existen otros procesos que están en constante movimiento dentro del sistema aguardando a que se les asigne una determinada tarea, a éstos se les conoce como demonios, que pueden ser de usuario o de sistema; los procesos terminados en espera de salida se conocen como zombie, éstos no están vivos, no emplean ningún recurso ni efectúan ningún trabajo, son procesos que han sido detenidos pero no se les puede dar la salida hasta que el sistema los reconozca como terminados.

Solaris cuenta con diferentes comandos para planificar la ejecución de los procesos; el comando *at*, por ejemplo, permite especificar cuándo han de ejecutarse órdenes determinadas, mientras el comando *crontab* planifica el tiempo exacto en que el demonio CRON ha de ejecutar trabajos específicos; por su parte, el comando *nohup* se emplea para asegurarse que un proceso siga vivo y se ejecute después de cerrar la sesión.

Para localizar procesos específicos se utilizan los comandos *pgrep* y *pkill*, el primero revisa la actividad de los procesos en el sistema y localiza los IDs de los procesos que se ajustan a los atributos especificados en la búsqueda, el segundo hace lo mismo, sólo que termina los procesos una vez que los ha ubicado.

Los equipos Sun y el sistema operativo Solaris ofrecen la posibilidad de tener varios usuarios dentro del equipo desempeñando determinadas tareas, para agregar un nuevo usuario al sistema se utiliza el comando *useradd*, para borrar algún usuario dentro del mismo se emplea el comando *userdel*. También es posible adicionar un grupo de trabajo con varios usuarios al sistema mediante el comando *groupadd*, para eliminar alguno de esos grupos se utiliza el comando *groupdel*.

Es importante que el administrador del sistema sepa definir las variables que el shell contiene y que permiten personalizar la información de directorios y archivos relativos a los programas, así como particularizar la forma en que el usuario interactúa con los programas.

Una de las tareas prioritarias del administrador es mantener organizado el sistema de archivos, para ello existen opciones para desplazar y renombrar archivos, como el comando `mv`, que permite mover uno o varios archivos de un directorio a otro, además de renombrar archivos y mover directorios con todos sus subdirectorios y archivos; también se pueden copiar archivos con el comando `cp`, de modo que la información se pueda modificar sin perder la original, este comando copia los archivos a otro destino.

También es importante que el administrador maneje el comando `vi`, que es una aplicación para la creación y modificación de textos; el editor `vi` tiene dos modos separados de ejecución: de entrada y de orden, además de tener diferentes herramientas para editar texto.

Para preservar el equipo y la información que contiene, el administrador del sistema debe manejar las diferentes herramientas de seguridad del sistema. El sistema Solaris cuenta con eficaces medidas de seguridad que restringen el acceso a usuarios no autorizados y protegen los recursos del equipo, sin que esto implique limitantes a los usuarios autorizados. Hay diferentes comandos de seguridad, como los archivos de contraseña (`/etc/passwd` y `/etc/shadow`) que guardan información de los usuarios y son utilizados por el programa `login` para validarlos y preparar el entorno de trabajo inicial.

Algunos de los comandos del sistema para aumentar la seguridad en la información son: `/etc/group` (define todos los grupos del sistema y especifica a cuáles pueden acceder los diferentes usuarios), `/var/adm/sulog` (guarda una bitácora de los usuarios que utilizaron el comando `su` para entrar al sistema como superusuario), `/etc/default/passwd` (especifica valores asignables a las contraseñas), `/etc/default/login` (restringe el acceso a la cuenta de superusuario), `/etc/default/su` (registra el acceso de todos los usuarios), `/etc/host.equiv` (lista las máquinas de confianza para el anfitrión y los usuarios remotos), y `.rhost` (controla el acceso de usuarios remotos a directorios específicos).

Para asegurar que los usuarios accedan a información correspondiente a su nivel, existen tres clases de permisos para archivos que corresponden a las clases de usuarios: el propietario (o usuario) del archivo, el grupo al que éste pertenece y los otros usuarios del sistema. El sistema operativo Solaris permite fijar los permisos para cada archivo, pero solamente el propietario del archivo puede modificarlos para que los archivos o directorios sean públicos, se compartan sólo con los miembros de un grupo, o bien sean privados; el comando `chmod` permite realizar tales cambios archivo por archivo. El comando `umask`, permite realizar automáticamente estos mismos cambios, pero con una sola orden.

Existen siete niveles estándar de seguridad en equipos computacionales, con niveles y subniveles de protección de la información y de los equipos; estos niveles son: protección mínima, de seguridad discrecional, de acceso controlado, de seguridad etiquetada, estructurada, dominios de seguridad y diseño verificado.

Además de las medidas de seguridad, el administrador del sistema debe conocer cómo manejar los dispositivos del equipo para que éstos funcionen perfectamente; así, por ejemplo, debe tener conocimientos sobre cómo adicionar, eliminar y configurar una impresora. El sistema operativo Solaris contiene una colección de programas para imprimir archivos y documentos (sistema `lp`) que proporciona una interfaz simple y uniforme para una gran variedad de impresoras; cuando se agrega al sistema una impresora, es preciso detener el planificador `lp`, identificar la impresora, reiniciar el planificador, notificar de la adición y activar la impresora.

Asimismo, el administrador deberá tener conocimientos sobre los usos, estructura y funciones de los file system. Un file system es una colección de archivos y directorios utilizados para colocar y organizar la información, en el sistema operativo es una colección de estructuras de control y bloques de datos que ocupan un espacio definido por una partición. Hay varios tipos de file system que se pueden implementar en discos duros, CD ROM o diskette, entre ellos se encuentran: `ufs`, `hfs`, `pcfs` y `RAM-based`.

El sistema Solaris cuenta con una serie de comandos que sirven para optimizar el funcionamiento de los file system, como el comando `fsck` (file system check), que verifica el estado de cada file system cuando el sistema operativo es iniciado, en busca de inconsistencias en su funcionamiento, para ello cuenta con dos diferentes modos: interactivo y no interactivo.

El administrador del sistema Solaris, se verá con frecuencia en la necesidad de montar file system, que es el proceso por el cual los file system son separados y adicionados a una estructura jerárquica; el montaje y desmontaje de file system ocurre durante el encendido y apagado del sistema por medio del comando `mount`. El archivo `/etc/vfstab` contiene las entradas por omisión del montaje de los file system al inicio del sistema, ya que durante el inicio o el apagado del sistema, todos los file system son montados o desmontados automáticamente.

Otro tipo de conocimientos esenciales para el administrador de Solaris, son los concernientes a la administración de discos dentro del sistema operativo. En los sistemas Sun, el almacenamiento en disco se divide en secciones (particiones) que se agrupan a su vez en cilindros, lo que facilita que el administrador organice funcionalmente los datos.

De igual manera, es conveniente que el administrador del sistema maneje las diferentes formas de nombrar a los dispositivos; Solaris se refiere a ellos de cuatro formas diferentes: lógica, física, instancia y BSD. El comando `dmesg` identifica los dispositivos conectados al sistema por medio de nombres de instancia y nombres de dispositivo físico; por su parte, el comando `format` despliega los nombres de dispositivo tanto lógicos como físicos.

El administrador debe conocer también el proceso de reparticionamiento, pues ocasionalmente es necesario reparticionar un disco, ya sea para volver a dimensionarlo o para volver a etiquetar un disco con una nueva tabla de partición; para evitar que la información de tales discos se pierda, es preciso respaldar los file system. El comando `format` es una herramienta de mantenimiento de disco que sirve para configurar los tamaños de las particiones del disco.

Una función indispensable en la administración del sistema es el respaldo de los archivos como medida preventiva en caso de pérdida o daños. Ésta es una de las funciones más importantes para el administrador del sistema, por ello los respaldos deben ser planeados y realizados periódicamente; existen tres tipos de respaldos: totales, incrementales y hechos en varios volúmenes. El `ufsdump` es un comando para respaldar el file system; este respaldo puede ser total, incremental o de archivos y directorios individuales, además, cuenta con varias opciones para adquirir propiedades extras en el respaldo.

Para restaurar la información de los respaldos se utiliza el comando `ufsrestore`, que extrae los archivos desde el respaldo y tiene opciones que proporcionan características adicionales para mejorar su uso. Otro comando para respaldar archivos y directorios pequeños es el comando `tar`, que respalda uno o varios archivos en un directorio jerárquico. Actualmente, se utiliza el comando `cpio` que tiene mayor flexibilidad que `tar`, pues dirige sus entradas y salidas a las estándar.

Una de las labores más importantes para el administrador del sistema es la de adicionar hardware al equipo, para lograrlo adecuadamente, debe seguir una serie de pasos, desde colocar el hardware, conectar físicamente el disco y reiniciar el sistema, reconocer el disco y particionarlo, crear los file systems, editar la tabla del archivo `/etc/vfstab` y reiniciar el sistema para que las modificaciones sean incorporadas la mismo.

Hay ciertos comandos de administración avanzada que el administrador del sistema operativo Solaris debe conocer, algunos de ellos son: `iostat` (anuncia la actividad del disco a la terminal y puede utilizarse para evaluar las cargas en cada disco del sistema servidor), `vmstat` (informa estadísticas sobre procesos, encolamientos, memoria virtual, discos, trap y actividades del CPU), `swap` (se emplea para extender el espacio y para agregar o mover áreas de swap), `netstat` (brinda información relacionada con la red, incluyendo estadísticas para cada nivel de red), `snoop` (captura y despliega el contenido de los paquetes en una red), `mpstat` (reporta estadísticas de información sobre el procesador), y `sar` (reporta la utilización de los recursos del sistema).

CAPÍTULO IV

IV INTRODUCCIÓN A NETBACKUP 3.2

Introducción

El siguiente capítulo es un valor agregado a esta tesis ya que tiene como objetivo el presentar una breve introducción al software llamado NetBackup 3.2 y mostrar características del dispositivo de almacenamiento Sun Storage L11000.

Para hacer el ejemplo más claro se tomaron datos verídicos de una empresa real donde tiene una política de respaldo basada en este software y hardware.

Es probable que en algún momento la persona que empieza a trabajar con la administración del sistema operativo Solaris necesite realizar respaldos de información, y tal vez con el tiempo requiera crear una política de respaldo más eficaz, implementar algún tipo de software que realice los backups automáticamente y algún tipo de hardware que permita realizar ese objetivo, de ahí la inquietud por dar a conocer a grandes rasgos las características principales de NetBackup y la Librería L11000.

4.1 Dispositivos de almacenamiento

Existen multitud de dispositivos diferentes para almacenar los datos del sistema, desde un disco flexible hasta unidades de cinta de última generación. Evidentemente, cada uno tiene sus beneficios y sus inconvenientes. En este apartado se van a comentar algunos de los dispositivos de almacenamiento que fueron o son los más utilizados hoy en día; de todos ellos (o de otros no comentados aquí), cada administrador ha de elegir el que más se adapte a sus necesidades.

Discos flexibles

“Aunque los clásicos *diskettes* cada día se utilicen menos, aún se pueden considerar dispositivos donde almacenar copias de seguridad. Se trata de un medio barato y portable entre diferentes sistemas operativos (evidentemente, esta portabilidad existe si utilizamos el disco como un dispositivo secuencial, sin crear sistemas de ficheros). Por contra, su fiabilidad es muy baja: la información almacenada se puede borrar fácilmente si el disco se aproxima a aparatos que emitan cualquier tipo de radiación, como un teléfono móvil o un detector de metales”¹⁸. Asimismo, la capacidad de almacenamiento de los disquetes es muy baja, de poco más de 1MB por unidad; esto hace que sea casi imposible utilizarlos como copia de seguridad de grandes cantidades de datos, restringiendo su uso a archivos individuales y pequeños.



Fig. 4.1 Los disquetes de 3 ½ son comúnmente utilizados para información individual.

¹⁹ Villalón Huerta, Antonio. *Seguridad en UNIX y redes*. p.106

Discos duros

Es posible utilizar una unidad de disco duro completa (o una partición) para realizar copias de seguridad; al igual que con los discos flexibles, se puede crear un sistema de archivos sobre la unidad o la partición correspondiente, montarla y copiar los archivos que nos interesan en ella (o recuperarlos). De la misma forma, también se puede utilizar la unidad como un dispositivo secuencial y convertirlo en un contenedor de tar o ufsdump.



Fig. 4.2 Un disco duro también se puede utilizar para realizar copias de seguridad

Cintas magnéticas

Las cintas magnéticas han sido desde hace muchos años el dispositivo de almacenamiento por excelencia. Las más antiguas, las cintas de nueve pistas (un elemento circular con la cinta enrollada en él) se utilizaron durante mucho tiempo, aunque en la actualidad se son inusuales, ya que, pese a ser altamente fiables y a su relativa velocidad de trabajo, tienen una capacidad muy limitada (las más avanzadas son capaces de almacenar menos de 300 MB, lo que resulta insuficiente en la mayoría de los sistemas actuales).

Después de las cintas de 9 pistas surgieron las de un cuarto de pulgada denominadas QIC (Quarter Inch Cartridge) "mucho más pequeñas en tamaño que las anteriores y con una capacidad máxima de varios Gigabytes (aunque la mayor parte de ellas almacenan menos de un Giga); se trata de cintas más baratas que las de 9 pistas, pero también más lentas. El medio ya no va descubierto, sino que va cubierto de una envoltura de plástico"²⁰.

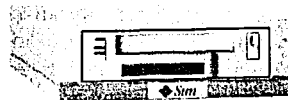


Fig. 4.3 Equipo lector de las cintas QIC.

A finales de los ochenta surge un nuevo modelo de cinta que desplaza las cintas QIC a un segundo término y que se ha vuelto el medio de respaldo más empleado en la actualidad; se trata de las cintas de 8mm, diseñadas originalmente para almacenar vídeo. Estas cintas, del tamaño de un cassette de audio, tienen una capacidad de hasta 5 GB, lo que las hace perfectas para la mayoría de sistemas: como todos los datos a respaldar caben en un mismo dispositivo, el administrador puede introducir la cinta en la unidad del sistema, ejecutar un sencillo shell, y dejar que el backup se realice durante la noche, al día siguiente bastará con verificar que no han ocurrido errores.

²⁰ Ídem, p. 108

Sin embargo, este tipo de cintas presenta una gran complicación: como se ha dicho, fueron originalmente diseñadas para almacenar vídeo, y se basan en la misma tecnología para registrar la información; el inconveniente radica en que, al grabar imágenes en una cinta de este tipo, no importa perder algunos bits de ella, pues aún así la imagen almacenada mantiene su sentido, por el contrario, perder esa misma cantidad de bits en una cinta donde se ha respaldado cierta información, resultaría lamentable, pues el resto del contenido de la cinta podría resultar inservible. Incluso, es posible que después de algunos usos (incluyendo las lecturas) la cinta se dañe irreversiblemente.

Para solucionar algunos de estos problemas se crearon las cintas DAT, de 4mm, diseñadas ya en origen para almacenar datos. Estos dispositivos, un poco más pequeños que las cintas de 8mm, cuentan con una capacidad similar, además son más recientes y baratas, por ello son el mejor sustituto de las cintas antiguas.



Fig. 4.4 Equipo lector de cintas DAT.

CD-ROM

“En la actualidad sólo se utilizan cintas magnéticas en equipos antiguos o al almacenar grandes cantidades de datos [...]. Hoy en día muchas máquinas Unix poseen unidades grabadoras de CD-ROM, un hardware barato y, lo que es más importante, que utiliza dispositivos de muy bajo coste y con una capacidad de almacenamiento suficiente para muchos sistemas: con una unidad grabadora, podemos almacenar más de 650 Megabytes en un CD-ROM”²¹.

Se pueden utilizar dispositivos CD-RW, análogos a los anteriores, con la ventaja de que permiten eliminar la información almacenada y reutilizar el mismo dispositivo (lo que es muy útil si se vuelven a utilizar uno o varios juegos de copias), o utilizar medios con mayor capacidad de almacenamiento (CD-ROM de 80 minutos, capaces de almacenar hasta 700 MB).



Fig. 4.5 Equipo Sun para CD ROM.

²¹ Ídem, p.109

DVD

Durante muchos años no fue fácil encontrar un estándar capaz de poner de acuerdo a los principales fabricantes de CD-ROM, vídeos VHS, laserdiscs y equipos musicales. La tecnología DVD (Digital Video Disc) ha resuelto este problema creando el estándar de almacenamiento digital.

La especificación DVD (Digital Vídeo Disc) o (Digital Versatile Disc), es un intento por unificar todos los estándares óptico-digitales de almacenamiento, es decir, cualquier sistema de grabación que almacene imágenes o sonido. DVD abarca todos los campos informáticos actualmente existentes, por lo que un disco puede utilizarse para almacenar películas, música, datos informáticos, e incluso juegos.

Una ventaja del DVD, en relación a los sistemas actuales, es su mayor velocidad de lectura, hasta cuatro veces más rápido que los reproductores de CD tradicionales, y su gran capacidad de almacenamiento, que varía entre los 4.7 y los 17 Gigas, es decir, el tamaño aproximado de 25 CD-ROM. Todo ello, en un disco DVD que, externamente, es exactamente igual que un CD tradicional. Esta elevada capacidad permite, no solo almacenar gran cantidad de información, aplicable a todo tipo de programas, sino también reproducir 133 minutos de vídeo con excelente calidad.

Otra característica es que gracias a la compatibilidad con los sistemas actuales, los lectores DVD-ROM son capaces de leer los CD-ROM y CD musicales que actualmente existen.

Existen también las grabadoras DVD, en dos versiones diferentes. Las grabadoras DVD-R que son equivalentes a las grabadoras CD-R actuales, mecanismos "write once" que permite escribir en un disco DVD en blanco una sola vez. Estos discos tienen una capacidad cercana a los 3 Gigas, aunque se acercaran a los 4.7. Las grabadoras DVD-RAM son discos DVD que pueden borrarse y escribirse múltiples veces. Su capacidad es de 2,6 Gigas.

El DVD puede ser un buen dispositivo de almacenamiento, sin embargo, en el ambiente de backups o copias de seguridad, aún le falta "madurar", es decir, ésta es una tecnología nueva que está empezando a extenderse y por tanto puede tener detalles, desconozco si existen unidades de almacenamiento (Storage Unit o librerías) que permitan una automatización de los respaldos con estos dispositivos; este medio tiene una capacidad de respaldo que supera a muchos dispositivos de almacenamiento, pero en empresas donde se manejan varios GB de información habría que realizar un análisis con diferentes dispositivos de almacenamiento como los cartuchos DLT para poder seleccionar el que más se ajuste a las necesidades del sistema.



Fig. 4.6 Equipos DVD.

Cartuchos de cinta DLT

El cartucho DLT es la solución ideal para las necesidades de almacenamiento de datos, su robusto diseño y empaque puede compensar con muchos años de calidad y de servicio confiable; su diseño y fabricación están orientados a su durabilidad, un cartucho de éstos puede tener un tiempo de vida hasta de 30 años y pasadas de cabeza superiores a 1 millón.

Este dispositivo de almacenamiento ofrece máxima confiabilidad, durabilidad y capacidad para datos de las aplicaciones, ha sido diseñado para un rendimiento superior con librerías y drives DLT. Sin embargo, un manejo inapropiado o falta de cuidado, pueden afectar el rendimiento de cualquier producto bien diseñado.

El cartucho DLT está elaborado para cubrir estándares exactos; más que una cinta en un carrete abierto, un cartucho es un componente complicado y de precisión del sistema DLT y contiene características especiales, como el reel locks (que sujeta la cinta en la tensión correcta cuando el cartucho de cinta no está insertado en el drive DLT). Las complicaciones suelen ocurrir cuando el cartucho de cinta se ha caído o dañado accidentalmente o a causa del stress ambiental.

A continuación se presenta una guía general de manejo e instrucción para la inspección visual simple de los cartuchos DLT, a modo de asegurar que éstos se encuentran en perfectas condiciones de almacenar y preservar los datos antes de ser utilizados.

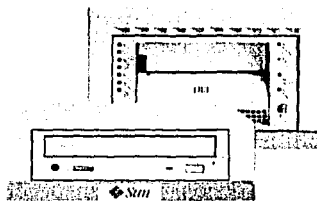


Fig. 4.7 Equipo Sun para cartuchos DLT y CD.

Guía general de manejo de los cartuchos DLT

Guardar siempre cada cartucho de cinta en su estuche de plástico protector cuando no esté dentro de la librería.

Al llevar cartuchos de cinta en sus estuches, orientar siempre los estuches de modo que las estrías se aseguren entre sí.

Nunca colocar los cartuchos de cinta en pilas de más de cinco.

Procurar siempre las condiciones de ambiente apropiadas para su almacenamiento.

Al poner los cartuchos en un anaquel para su almacenamiento, asegurarse de poner cada cartucho en forma vertical.

Evitar poner el cartucho de cinta cerca de cualquier fuente de campos magnéticos de alta intensidad, como monitores.

Nunca colocar etiquetas adhesivas en el cartucho de cinta, utilizar solo las etiquetas deslizables que vienen con él.

Nunca llevar los cartuchos en cajas o en cualquier otro contenedor, de modo que el cartucho vaya suelto, permitiendo que los cartuchos se golpeen entre sí o se expongan a golpeteos innecesarios.

No tocar o permitir el contacto directo con la cinta del polvo o la grasa natural de la piel, pues puede contaminar la cinta y afectar su rendimiento.

No exponer el cartucho de cinta a la humedad o a la luz directa.

Un cartucho que se ha caído puede tener sus componentes internos desensamblados o dañados, por ello, no se deben insertar en el drive cartuchos que se han caído o golpeado, sin que por lo menos se inspeccionen visualmente como se describe a continuación.

Procedimiento de inspección visual del mecanismo

La inspección visual del mecanismo del cartucho DLT debe ser una práctica habitual al momento de cambiar un cartucho o de cargar uno nuevo. Asimismo, si un cartucho de cinta se ha caído o fue golpeado fuertemente, o si el drive del DLT se hace inoperable después de cargar un cartucho de cinta, o bien, si se recibe un embarque de cartuchos de cinta con visibles muestras de que el empaque está dañado, es conveniente proceder a una inspección visual de los cartuchos.

Inspección visual de un cartucho de cinta DLT

Sacar el cartucho de cinta de su estuche protector de plástico.

Observar el cartucho de cinta para verificar si presenta alguna ruptura obvia o cualquier daño físico.

Agitar suavemente el cartucho de cinta, escuchar si hay piezas sueltas en su interior si se oye algo suelto dentro del cartucho, no se utilice.

Siguiendo los procedimientos de manejo general, haciendo cuidadosas inspecciones visuales del cartucho en una forma habitual y asegurándose de almacenar los cartuchos dentro de los límites ambientales, se reducirán al mínimo problemas con los cartuchos o posibilidades de dañar los drives DLT, protegiendo así los medios y los datos.

4.2 Copias de seguridad

Los procedimientos de copia de seguridad y restauración tienen una importancia capital, pues protegen de la pérdida de datos valiosos en la computadora.

La copia de seguridad es un procedimiento para copiar datos del sistema o información de particionamiento desde el medio de almacenamiento permanente de la computadora (generalmente un disco fijo) hasta otro medio (por lo común un medio extraíble como un disquete flexible o una cinta). La información de particionamiento describe las áreas diferentes del disco sobre las cuales están almacenadas las diferentes clases de datos. La restauración es un procedimiento para devolver al sistema versiones de los archivos, sistemas de archivos o información de particionamiento a partir de la copia de seguridad.

Una buena estrategia de copia de seguridad asegurará que pueda recuperarse una versión anterior de un archivo en caso de que éste sea borrado o modificado. También permitirá restaurar los archivos de configuración del sistema si resultan dañados o son destruidos.

“Las copias de seguridad del sistema son con frecuencia el único mecanismo de recuperación que poseen los administradores para restaurar una máquina que por cualquier motivo [...] ha perdido datos. Por tanto, una correcta política para realizar, almacenar y en caso de ser necesario, restaurar los backups es vital en la planificación de seguridad de todo sistema”²².

Relacionados a las copias de seguridad existen algunos problemas comunes; por ejemplo, el no verificar los backups realizados; en ocasiones el administrador ha diseñado una política de backup correcta, incluso exhaustiva, pero nadie se ocupa en verificar estas copias hasta que es necesario restaurar alguna información de ellas. Obviamente, cuando esto se presenta, el responsable del sistema se enfrenta a un gran problema: ¿qué pasaría si el respaldo está dañado?; este problema se pudo haber evitado teniendo la precaución de supervisar el correcto funcionamiento de los respaldos. Ciertamente, verificar una copia completa para corroborar que todo es correcto, puede ser demasiado trabajo para los métodos habituales de operación, lo usual es tratar de recuperar aleatoriamente varios archivos del respaldo, dando por hecho que si esta recuperación funciona, toda la copia es correcta.

Otro de los problemas más comunes de los backup es la política de etiquetado a seguir, pues, a pesar de que es una actividad muy útil para la ubicación de los respaldos, existen algunos administradores que no etiquetan sus dispositivos de almacenamiento, lo que representa inminentes dificultades, pues, cuando se presenta el momento de recuperar archivos, el operador ha de ir cinta por cinta o disco por disco tratando de averiguar dónde se encuentran las últimas versiones de tales archivos.

Por otro lado, existen confusiones sobre la correcta forma de etiquetar los backups; son muchos los administradores que siguen una política de etiquetado muy grande, proporcionando todo tipo de detalles sobre el contenido exacto de cada dispositivo, esto, que en principio puede parecer una posición correcta, no lo es, si por alguna razón alguien desea sustraer cierta información, no tiene que investigar mucho para conocer el contenido de cada cinta, lo que le proporcionaría acceso a información valiosa.

²² Ídem, p.105

Una política correcta para etiquetar los backups es la que permita al administrador conocer la ubicación y situación exacta de cada cinta, esto se consigue, por ejemplo, con códigos impresos en cada etiqueta, cuyo significado sea conocido por los operadores de copias de seguridad pero no por una persona ajena. Un adecuado etiquetado permite ordenar y controlar las copias de seguridad y la información que contienen, así como garantizar la protección de los datos almacenados

“La ubicación final de las copias de seguridad también suele ser errónea en muchos entornos, generalmente, los operadores tienden a almacenar los backups muy cerca de los sistemas, cuando no en la misma sala”²³. Esta estrategia de almacenamiento, que ofrece mayor comodidad a los operadores para recuperar información, puede representar graves complicaciones; así, por ejemplo, puede ocurrir cualquier desastre, como un incendio o robo de los equipos dando por resultado la pérdida de información.

Por otro lado, existe quien afirma como correcto el almacenar las copias de seguridad junto a los propios equipos, ya que así se consigue centralizar un poco la seguridad (protegiendo una única estancia que guarda tanto los equipos como las copias). Lo recomendable en cualquier empresa es poder tener un juego de copias de seguridad completas en un lugar diferente a la sala de operaciones, pero protegido y un juego para uso diario en la propia sala, de manera que los operadores puedan recuperar archivos fácilmente.

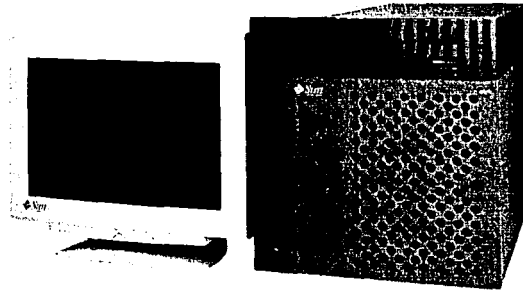


Fig. 4.8 Sun Enterprise 450.

¿Por qué hacer respaldos?

- Pueden suceder equivocaciones que, para resolverse, necesiten una restauración de datos.
- Cuando ocurre una pérdida de datos en los arreglos de discos.
- Las cintas son relativamente más baratas que los discos.
- El volumen de datos crece rápidamente.

²³ Íbidem.

4.3 Sun StorEdge Enterprise NetBackup

Sun StorEdge Enterprise NetBackup provee un alto rendimiento en la ejecución de respaldos, archivado y servicios de restauración para clientes y servidores en redes cliente/servidor. El rango de operación puede hacerse desde un sistema standalone a uno entiere enterprise. El hardware puede ser plataforma UNIX, Microsoft Windows, OS/2, y Macintosh. Para este trabajo solo me enfoco a UNIX Solaris.

Los dispositivos de almacenamiento pueden ser discos, cintas u ópticos. El manejador de medios de Sun StorEdge Enterprise NetBackup maneja cintas y dispositivos ópticos de almacenamiento y está diseñado para que otros dispositivos secundarios de almacenamiento puedan ser compartidos por otros productos Sun StorEdge.

El administrador puede colocar periodicamente calendarios para automatizar los respaldos, atender operaciones de respaldo para clientes por red, realizar solo respaldos de archivos recientemente modificados. Teniendo cuidado al realizar los calendarios de automatización de los respaldos el administrador puede archivar sistemáticamente los respaldos sobre un periodo de tiempo y optimizar el tráfico en la red durante las horas de mayor carga.

En relación al calendario de los respaldos, un respaldo manual de datos solicitado por un cliente puede ser ejecutado usando el mismo criterio especificado para un respaldo automático, es decir, si ya existe un respaldo de datos automatizado puede ser lanzado manualmente. La operación de un respaldo manual es ejecutada en circunstancias especiales, como la solicitud de un usuario que ha realizado cambios muy importantes dentro de sus archivos o para preservar la configuración del sistema antes de actualizar algún tipo de software.

Los usuarios pueden ser capacitados para realizar las tareas de archivar, respaldar y recuperar la información sin la intervención de un administrador.

NetBackup mantiene una base de datos en el servidor con todos los archivos de información de los respaldos y restauraciones. Separando los procesos especiales, proporcionando facilidad para recuperar información en caso de alguna pérdida.

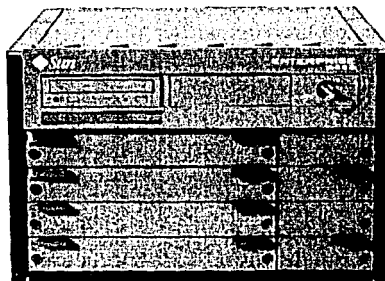


Fig. 4.9 Sun Enterprise 4500.

Características de NetBackup

- Posee alta ejecución en respaldo, archivado y recuperación de información.
- Soporta un medio ambiente cliente/servidor en red.
- Tiene gran escalabilidad desde un sistema Standalone hasta un Entire Enterprise.
- Soporta UNIX, Windows, OS/2 y Macintosh.
- Soporta discos, cintas, robots, discos ópticos así como dispositivos de almacenamiento.
- Puede realizar respaldos de cualquier cliente en la red.
- Posee interfaz de administración para automatizar procesos.
- Los clientes pueden iniciar respaldos y restauraciones en su sistema.
- NetBackup tiene un catálogo de todos los respaldos y restauraciones.

NetBackup servidor y clientes

Sun StorEdge Netbackup incluye el software para clientes y servidor. El software del servidor reside solo en la plataforma que manejará los dispositivos físicos utilizados para almacenamiento secundario. El software cliente reside en una estación de trabajo individual conteniendo los datos que van a ser respaldados (un servidor puede ser un cliente). En esta arquitectura, el software cliente es responsable de generar la cadena de datos a respaldar y el software servidor dirige esta cadena al dispositivo de almacenamiento secundario.

Sun Storedge Enterprise Netbackup: Manejador centralizado

Sun Storedge Enterprise Netbackup acomoda múltiples servidores trabajando juntos en un backup cluster, bajo el control administrativo de un servidor en el cluster. En esta relación el controlador de la administración es designado como el servidor "maestro," y el otro servidor en el cluster, es designado el servidor "esclavo", operando bajo el control del servidor maestro.

Todas las funciones de administración son ejecutadas desde una localidad en el servidor maestro, y el servidor maestro controla todos los respaldos planificados para cada esclavo. Cada esclavo ejecuta una operación de respaldo bajo la dirección de maestro y realiza el respaldo local de sus datos en los dispositivos que tiene asignados.

Netbackup cluster provee muchas ventajas entre ellas, alta disponibilidad que asegura operaciones continuas en el medio ambiente maestro/esclavo. Por ejemplo, si una unidad de almacenamiento (robot o dispositivo de cintas) en un servidor de Netbackup dentro del cluster se "cae", el servidor maestro puede automáticamente dirigir los respaldos a otra unidad en el cluster. Si un esclavo sufre un crash, otro servidor esclavo puede tomar las operaciones de respaldo. Todas estas ventajas aseguran que los datos en el cluster sean protegidos y rápidamente recuperados en cualquier evento de falla que se presente.

4.4 Introducción a la librería Sun StorEdge L11000

La librería Sun StorEdge L11000, es un equipo automatizado de respaldo y recuperación, consta de hasta 16 drives de cinta de 326 slots para cartuchos DLT.

Cuando se llena totalmente con cartuchos, la librería tiene una capacidad máxima de 11.4 Terabytes (11.410 GB), basada en 326 cartuchos de 35 GB cada uno.

La librería almacena cartuchos de cintas en las siguientes ubicaciones:

170 storages bins en la pared interna, storage bin 000 al 169, tiene cinco columnas de 32 storages bins cada una y dos columnas en los extremos de cinco storages bins cada una.

60 storages bins en la puerta derecha, storage bin 170 al 229, tiene tres columnas en la parte superior de 4 storages cada una y tres columnas en la parte inferior de 16 storages cada una.

96 storage bins en la puerta izquierda, storage bin 230 al 325, tiene tres columnas de treinta y dos storages bins cada una.

Contiene un load port con dos magazines de 6 cartuchos cada uno y cuenta también con 16 drives DLT 7000.

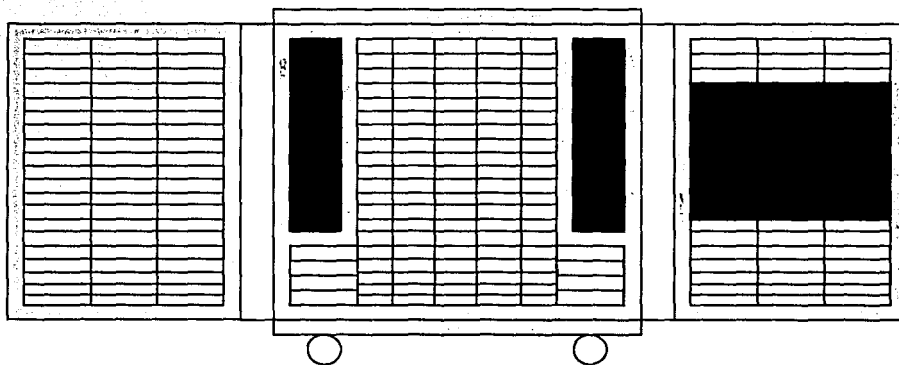


Fig. 4.10. Interior de la librería L11000.

Características de la Librería

Alta capacidad, elevado rendimiento de respaldo y recuperación.

Hasta 5 librerías pueden unirse en una unidad virtual.

Tecnología que asegura compatibilidad con futuras tecnologías.

Módulos de fuentes de poder redundantes hot-swappable, que aseguran la operación de la librería contra fallas en las fuentes de poder.

Sistema avanzado de enfriamiento para prevenir fallas causadas por sobrecalentamiento.

Intercambio de cartuchos On-line con un puerto de carga con 2 magazines de 6 cartuchos cada uno.

Facilidad para su servicio.

Drives DLT, fuente de poder y ventiladores son hot-swappable, que permiten a los ingenieros de servicio realizar reparaciones sin tener que apagar la librería.

Fácil acceso a componentes críticos que son reemplazables.

Panel de control touch screen con amplio rango de configuraciones y funciones relacionadas con servicio.

Componentes de la librería

Gabinete (Fig. 4.11).

Ventana (1).

Puertas (2).

Interfaz gráfica de usuario (GUI) touch screen (3).

Switch de encendido. (4)

Puerto de carga (5).

Mecanismo de manejo de cartuchos IntelliGrip (interno).

Drives de cinta (interno).

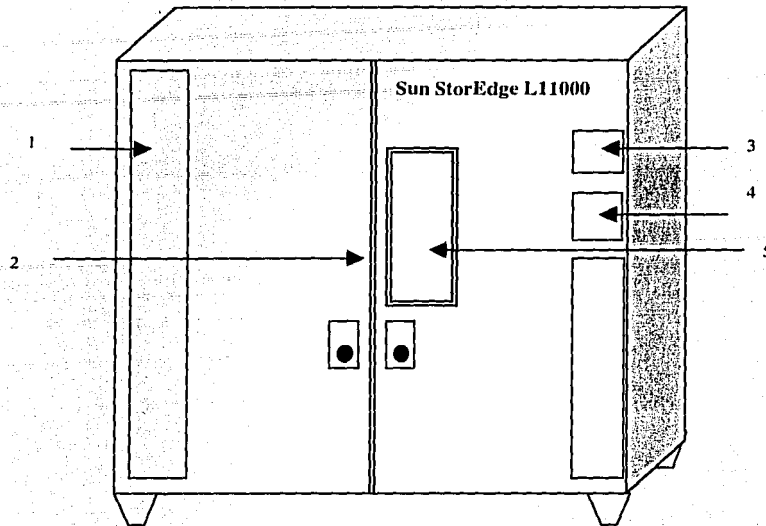


Fig. 4.11 Gabinete de la Librería L11000.

El gabinete aloja todos los componentes de la librería formando así un equipo automático de respaldo y recuperación de información. Si a este equipo adicionamos una herramienta de software de respaldo como Netbackup y una óptima política de respaldo se podría asegurar que siempre se tendría la información disponible en caso de alguna contingencia.

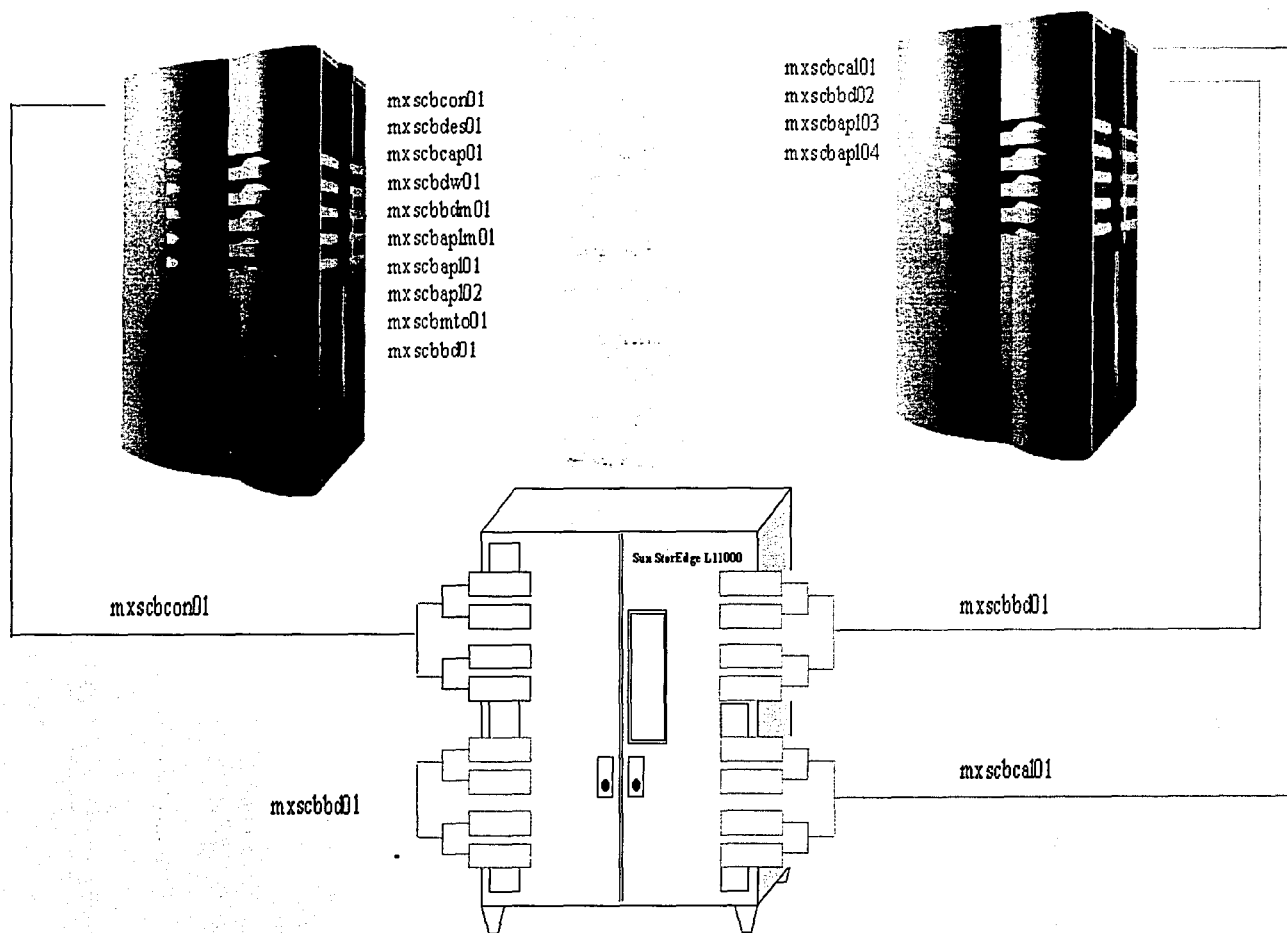
4.5 Ambiente

El propósito de este apartado es dar a conocer un breve ejemplo real de una empresa donde se maneja el sistema operativo Solaris y el software NetBackup 3.2 para llevar a cabo los respaldos de información.

En la empresa se cuenta con un dominio (mxscbcon01) con Solaris 2.6, que será el servidor de respaldo como NetBackup Master Server con 4 drives de la librería conectados a éste (Fig. 4.12).

Un segundo dominio (mxscbcal01) con Solaris 2.6 que será un servidor esclavo con 4 drives de la librería.

Un tercer dominio (mxscbbd01) con Solaris 2.6 que será un servidor esclavo con 8 drives de la librería.



4.12 Diagrama de asignación de los drives DLT.

Una Librería L1 1000 con 16 drives DLT7000 y 326 slots para cartuchos de cintas.

Los clientes están en dos Sun Ultra Enterprise 10000 con 14 dominios.

Se tiene una red de administración y respaldos (10.20.20.XX).

Características del servidor Maestro (mxscbcon01)

Hostname: mxscbcon01
IP address: 10.20.20.30
Plataforma Sun Enterprise 10000
Tamaño de memoria 8192 Mb
Procesadores 7 a 400 MHz.

Características del servidor esclavo (mxscbcal01)

Hostname: mxscbcal01
IP address: 10.20.20.50
Plataforma Sun Enterprise 10000
Tamaño de memoria 8192 Mb
Procesadores 6 a 400 MHz.

Características del servidor esclavo (mxscbbd01)

Hostname: mxscbbd01
IP address: 10.20.20.33
Plataforma Sun Enterprise 10000
Tamaño de memoria 6144 Mb
Procesadores 12 a 400 MHz.

Características de la Librería L11000

Nombre Storage Device	L11000
Número de drives DLT7000	16
Número de Storage Bins	326
Número de Magazines en Load Port	2 magazines de 6 slots c/u.

¿Cómo iniciar NetBackup 3.2?

Para poder iniciar NetBackup 3.2 es necesario que estén corriendo los demonios correspondientes, para iniciar los demonios se debe ejecutar el shell script:

```
/usr/opensv/netbackup/bin/goodies/S77netbackup
```

Estos procesos se pueden monitorear con el comando:

```
# bpps -a
```

Para terminar los demonios de NetBackup se ejecuta el comando:

```
/usr/opensv/netbackup/bin/goodies/K77netbackup
```

Si se quiere iniciar y terminar los demonios al tiempo del boot del servidor de respaldos o al momento de hacer shutdown, los archivos se deben copiar a:

```
cp /usr/opensv/netbackup/bin/goodies/S77netbackup /etc/rc2.d
```

```
cp /usr/opensv/netbackup/bin/goodies/K77netbackup /etc/rc0.d
```

Para desplegar el ambiente gráfico de NetBackup se debe ejecutar el siguiente comando:

```
# xnb &
```

El comando anterior hará que se despliegue la siguiente pantalla, desde donde se puede elegir cualquiera de las 5 opciones de NetBackup (Fig. 4.13).

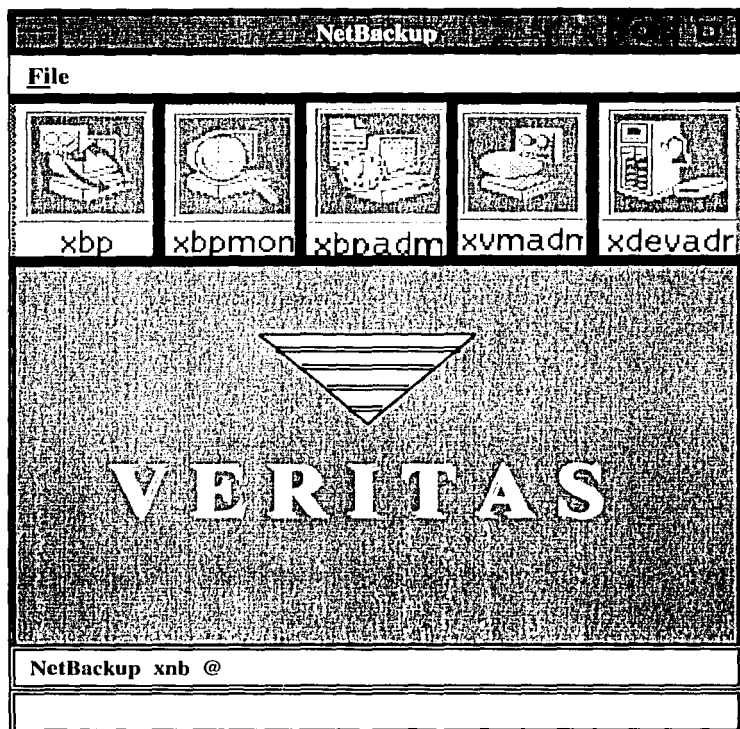


Fig. 4.13 Ventana principal de NetBackup.

Los menús principales de NetBackup son cinco:

xbp En este menú el cliente puede hacer respaldos y recuperación de datos. El icono es el siguiente:



xbpmon En este menú se monitorean los procesos de respaldo y recuperación en el momento en que se están ejecutando. El icono es el siguiente:



xbpadm En esta pantalla se pueden hacer configuraciones de clases, dar de alta a clientes o storage units. El icono es el siguiente:



xvmadm En esta pantalla se pueden administrar los medios magnéticos como cintas. El icono es el siguiente:



xdevadm En esta pantalla se pueden configurar dispositivos como robots y drives. El icono es el siguiente:



Para poder configurar una política de respaldo es imprescindible crear clases. Una clase es un conjunto de información para realizar respaldos de información automáticamente. La información que la clase puede contener es: nombre, calendario (cada cuando se realizará el respaldo), información a respaldar, de dónde se va a tomar y en dónde se va a depositar, entre otros datos.

Para crear una nueva clase en la ventana del administrador (xbpadmin), se debe oprimir en el menú *Actions* y dentro de este *new class*, como se muestra a continuación.

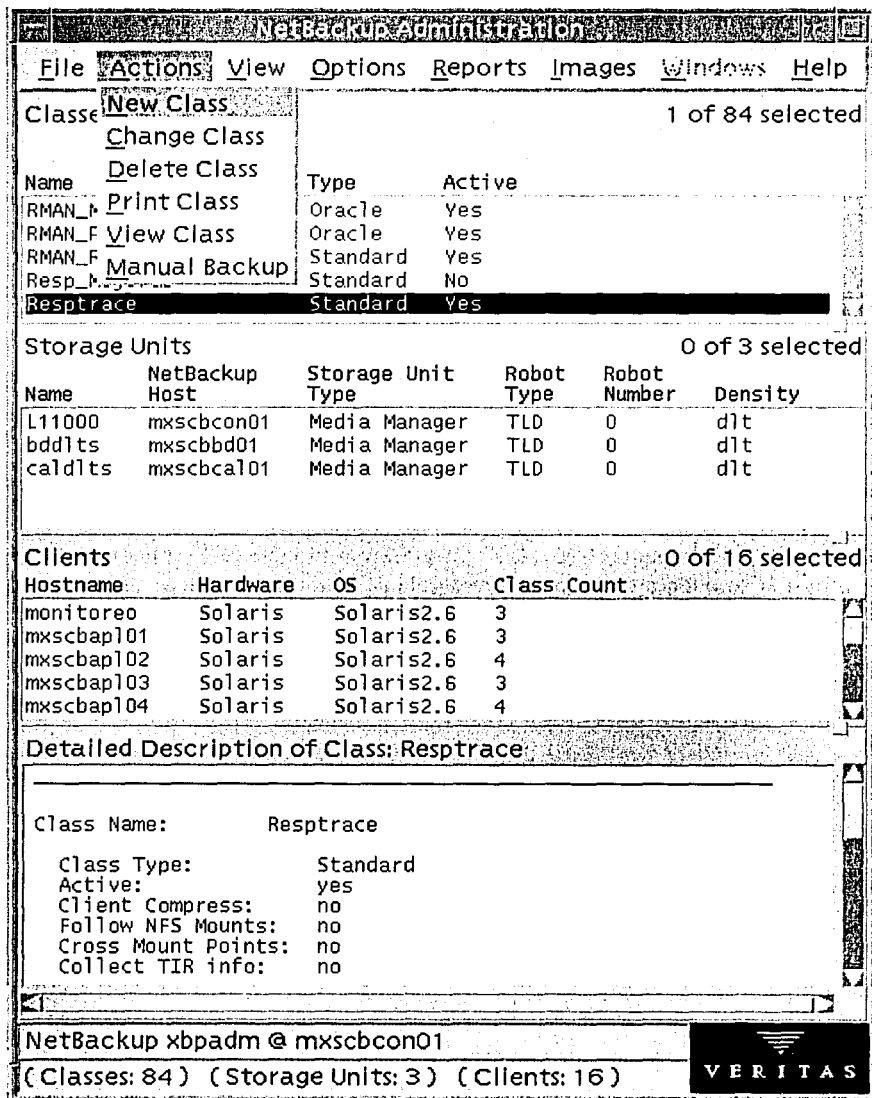


Fig. 4.14 Creación de una nueva clase

TESIS CON
FALLA DE CUBIEN

Ya que se ha realizado el paso anterior, aparecerá la siguiente ventana.

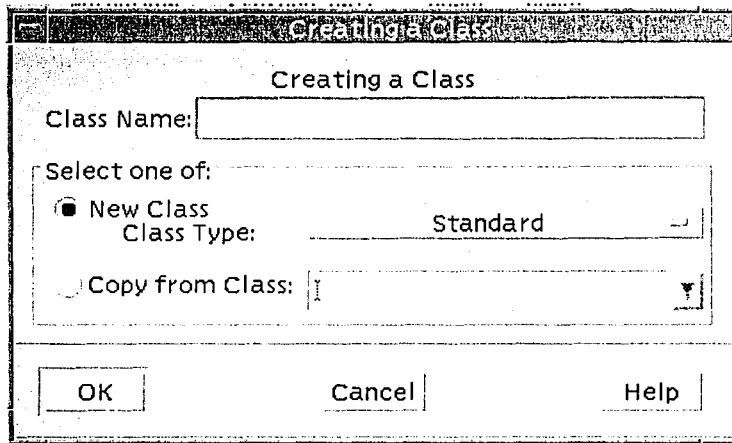


Fig. 4.15 Nueva clase.

Una vez que se presenta la pantalla se debe teclear el nombre que llevará la clase, es posible crearla desde el inicio o si se cuenta con una clase semejante a la que se quiere crear se puede oprimir el botón *copy from class*, esto presentará una lista con las clases ya existentes, si no se optó por este camino, se oprime el botón *ok* después de introducir el nombre y aparecerá la ventana siguiente:

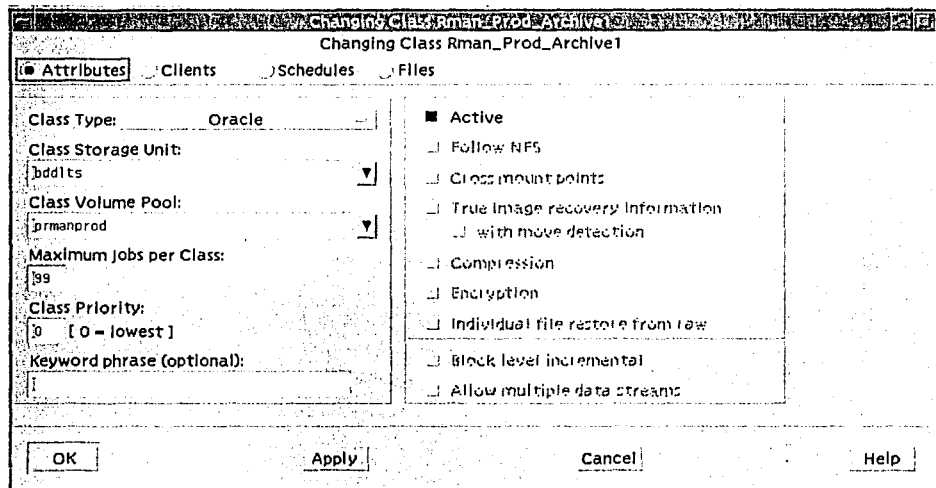


Fig. 4.16 Atributos.

Al aparecer la ventana anterior se pueden observar en la parte superior cuatro principales menús, a los cuales se les puede definir como las partes de una clase:

- ❖ Attributes
- ❖ Clients
- ❖ Schedules
- ❖ Files

En Attributes se puede definir:

- Tipo de clase, si es standard o de Oracle o Informix, etc.
- El storage unit donde se hará el respaldo.
- El pool de volúmenes para esta clase.
- El máximo número de trabajos para la clase.
- Si se define prioridad para la ejecución de esta clase.
- Una frase opcional para búsqueda de respaldos.
- Si está o no activa la clase.
- Si se desea utilizar compresión de datos.

En Clients, se despliega la pantalla con lo que se puede adicionar (New), copiar (copy) o borrar (Delete) clientes para esta clase. La pantalla mostrada es la siguiente:

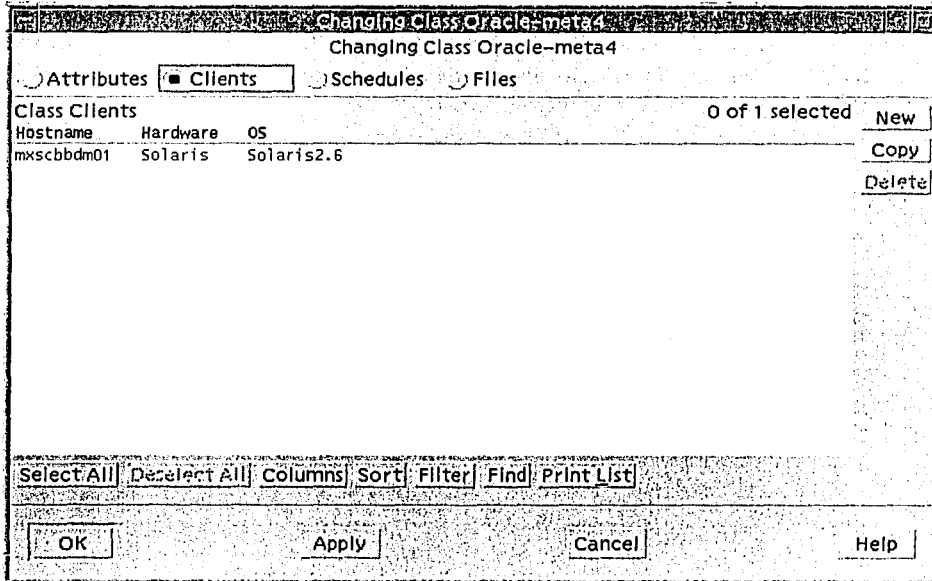


Fig. 4.17 Clientes.

Si se oprime el botón de *New*, se abrirá la ventana de *Adding Clients*, en la que se debe definir el hardware, el sistema operativo y cuál es el nombre; éste tiene que existir en el archivo *hosts* del servidor, aunque no se verifique en el momento. Es posible que una clase pueda tener múltiples clientes definidos. La ventana es la siguiente:

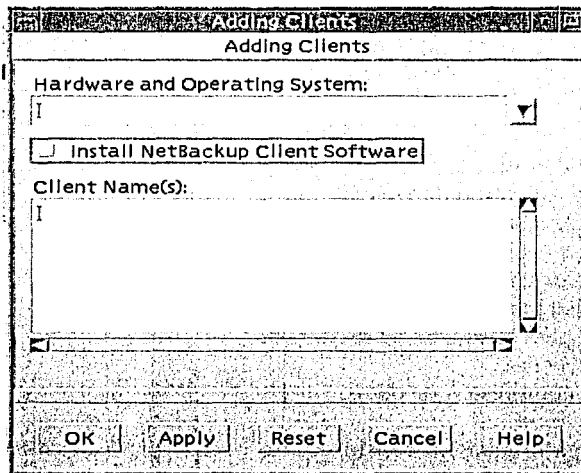


Fig. 4.18 Adicionar clientes.

En el menú *File* se puede definir una lista de archivos que se desea respaldar, pudiendo usar metacaracteres. La ventana es la siguiente:

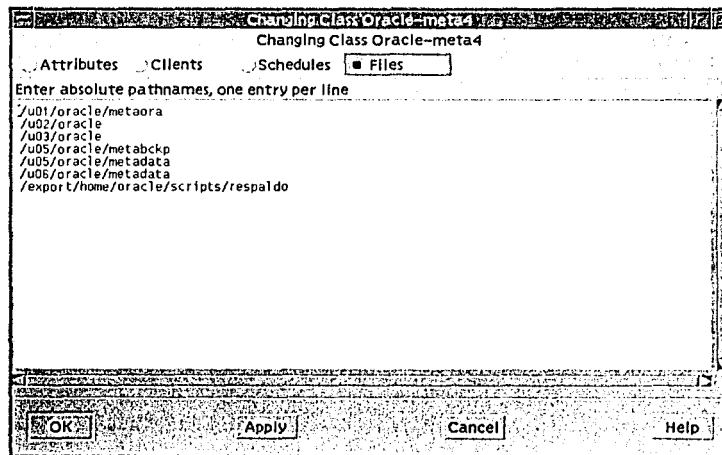


Fig. 4.19 Files.

En el menú Schedules aparecerá la siguiente ventana donde muestra un título de *changing class*, en la parte de la derecha existe un botón llamado *new*, que cuando se oprime, despliega el dialogo *creating a schedule* que, como el nombre lo indica, crea un nuevo calendario para una clase.

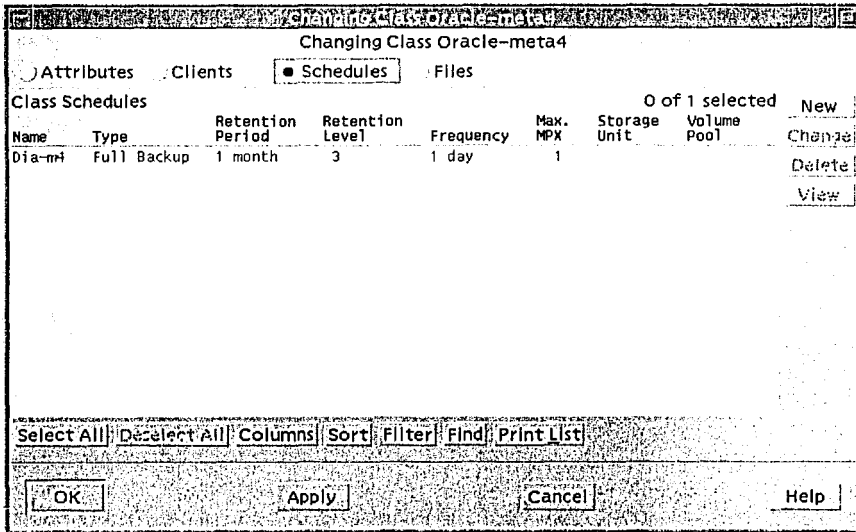


Fig. 4.20 Schedule.

Para crearlo se puede copiar un schedule de otra clase ya existente o crear uno nuevo tecleando el nombre que llevará. Aparecerá la siguiente ventana:

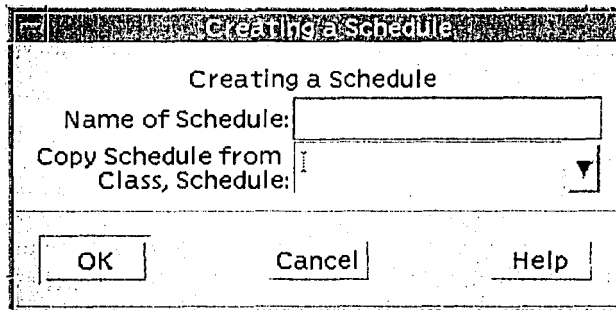


Fig. 4.21 Nuevo calendario.

Changing Schedule Dia-m4 of class Oracle-meta4

Schedule Name: Dia-m4

Override Class No
Storage Unit: Yes

Override Class No
Volume Pool: Yes

Type of Backup: Full Backup

Retention Period: 1 month

Frequency: 1 Days

Maximum MPX per Drive: 1

Schedule Times	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
Start:		23:55:00	23:55:00	23:55:00	23:55:00	23:55:00	
Duration:		06:00:00	06:00:00	06:00:00	06:00:00	06:00:00	
Ends:		Tuesday 05:55:00	Wednesday 05:55:00	Thursday 05:55:00	Friday 05:55:00	Saturday 05:55:00	

Clear Duplicate Copy

OK Cancel Help

Fig. 4.22 Opciones de schedule.

En la ventana anterior se puede definir el tipo de respaldo, el periodo de retención, la frecuencia con que se hará el respaldo, la hora y el tiempo que se le permitirá como límite para iniciar dicho respaldo.

Una vez concluida la definición de la clase, se puede iniciar un respaldo manual o calendarizado, de acuerdo a la definición del horario.

Es importante mencionar que este capítulo es solo una introducción para dar a conocer este software y hardware, que puede ser implementado como parte de un sistema con SO Solaris y crear una eficiente política de respaldo y recuperación de datos. En relación a ellos existen muchos temas afines de los cuales se podría llegar a realizar un trabajo muy vasto. La instalación de NetBackup y sus diferentes tipos, la configuración, el tipo de implementación, la instalación de la Librería Sun Storage L11000, configuración, etc, son temas que están abiertos para que alguien interesado en ellos pueda realizar algún trabajo particular de tesis.

Política de respaldo

Una de las funciones más relevantes del administrador de sistemas es mantener siempre disponible la información almacenada en él, para conseguirlo, generalmente se realizan respaldos de datos previniendo su pérdida a causa de caídas del sistema, eliminación de archivos accidentalmente, fallas de hardware, desastres naturales y posibles problemas al reinstalar o actualizar el sistema.

Es importante conocer el sistema que se está administrando para identificar cuáles son los file systems que contienen la información más importante para los usuarios y conocer cuáles son los sistemas de archivos que cambian más frecuentemente, para poder realizar una adecuada planeación de los respaldos.

Una política de respaldo es un grupo de información que permite crear un calendario eficaz en la realización de backups, en ella se pueden encontrar las cintas asignadas a cada respaldo, cuándo se realiza, a qué hora y qué información respalda.

Una política de respaldo es un proceso que depende de la actividad del sistema, por ejemplo: un equipo al que se le agreguen usuarios constantemente necesitará que se respalde con mayor frecuencia a nivel sistema operativo; en cambio, si el servidor contiene una base de datos, es probable que los respaldos de sistema operativo no sean tan necesarios como los respaldos a nivel base de datos. Cada sistema es diferente, por ello, crear una política eficaz depende de las necesidades de cada sistema, aún así, a continuación se presentan algunas sugerencias para respaldar sistemas de archivos del sistema operativo.

Partición de root: El file system raíz contiene el Kernel y puede contener también el directorio /var en el cual frecuentemente se modifican archivos de mail. Esta partición puede ser respaldada regularmente, una vez a la semana.

Partición /usr: cuando se adiciona nuevo software o comandos, generalmente son alojados en este directorio por lo cual se recomienda respaldarlo ocasionalmente, una vez a la semana.

Partición /export/home: contiene directorios y subdirectorios de todos los usuarios del sistema, este file system puede respaldarse una vez al día.

/export y /var pueden ser creados durante la instalación del sistema operativo, y pueden ser respaldados dependiendo de las necesidades.

En equipos donde se realizan diariamente pocos cambios dentro del sistema operativo, se recomienda ejecutar respaldos incrementales, los cuales tomarán los archivos que sufrieron alguna modificación a partir del último respaldo realizado.

Los comandos ufsdump y ufsrestore son los más recomendados para el respaldo y recuperación de datos ya que poseen varias cualidades como:

- El dispositivo de cinta puede estar en cualquier equipo dentro de la red en el cual el usuario tenga acceso.
- Pueden realizar respaldos incrementales de información.
- Puede ejecutarse el comando `ufsdump` remotamente, desde cualquier equipo en la red, siempre y cuando se tengan los privilegios suficientes.
- Se puede utilizar el comando `crontab` para ejecutarlo automáticamente.

Como se mencionaba anteriormente, crear una política de respaldo depende de las necesidades del sistema y es difícil sugerir alguna que pudiera acomodarse a las necesidades del común de los equipos, todos son diferentes en tamaño y necesidades, tienen diferente número de usuarios, aplicaciones, tareas específicas, etcétera.

Una de las ventajas del sistema operativo Solaris es que con las herramientas que proporciona se pueden crear políticas de respaldo acordes a los requerimientos específicos de información, y pueden ser modificadas en un momento determinado.

A continuación se presenta el ejemplo de la política de respaldo que se utiliza en la empresa, en ella se encuentran los nombres de los dominios que se respaldan, el nombre de la clase que respalda el dominio, ya sea a nivel base de datos o sistema operativo, incluyendo éste al propio sistema operativo y file systems, también contiene los días en que se respalda, la hora y el drive de la librería en donde se realiza el backup.

Es recomendable poner dentro de la política las cintas que están asignadas a cada respaldo y el orden de asignación dentro de la librería para una mayor localización de dónde se encuentran los respaldos, pero, el asignar un número definido de cintas a un respaldo puede ser algo problemático ya que este sistema cuenta con trece bases de datos, cada una de las cuales tiene asignadas varias cintas, sin embargo, estas bases de datos están en constante crecimiento por lo que en algún momento puede llegar a necesitarse un mayor número de cintas para el backup.

Para reducir el índice de dicho problema existe dentro de la librería un conjunto de cintas denominado *pool scratch* que permite asignar cintas automáticamente a cualquier clase cuando el proceso de respaldo así lo requiera.

Cabe mencionar que quedan dos drives libres de la librería de reserva en caso de que sea solicitado un respaldo manual o si se dañase alguno de los asignados.

Dominio	Clases	Nivel	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Semanal	Duración del respaldo diario	Duración del respaldo semanal	Aplicaciones por dominio del respaldo diario	Aplicaciones por dominio del respaldo semanal
Myscdsc01 (1)	ORACLE-MTRO (BD) SEMCOND1 (SO)	0	BD 21:00 DRIVE 0	BD 21:00 DRIVE 0	BD 21:00 DRIVE 0	BD 21:00 DRIVE 0	BD 21:00 DRIVE 0	BD 21:00 DRIVE 0	BD SO 21:00 DRIVE 0	3:30 h	5:00 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /db5/oracle /exp/eflowme /sysfiles	/usr /opt /netbackup /admin /patches /vaz
Myscdsc01 (2)	ORACLE-MEJS (BD) SEMDES01 (SO)	0	BD 21:00 DRIVE 1	BD 21:00 DRIVE 1	BD 21:00 DRIVE 1	BD 21:00 DRIVE 1	BD 21:00 DRIVE 1	BD 21:00 DRIVE 1	BD SO 21:00 DRIVE 1	2:30 h	3:15 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /db5/oracle /exp/eflowme /sysfiles	/usr /opt /admin /patches
Myscdsc01 (3)	ORACLE-CAPA (BD) SEMCAPO1 (SO)	0	BD 21:15 DRIVE 2	BD 21:15 DRIVE 2	BD 21:15 DRIVE 2	BD 21:15 DRIVE 2	BD 21:15 DRIVE 2	BD 21:15 DRIVE 2	BD SO 21:15 DRIVE 2	2:30 h	3:15 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /db5/oracle /exp/eflowme	/usr /opt /admin /patches
Myscdsc01 (4)	ORACLE-DW (BD) SEMDWO1 (SO)	0	BD 21:15 DRIVE 3	BD 21:15 DRIVE 3	BD 21:15 DRIVE 3	BD 21:15 DRIVE 3	BD 21:15 DRIVE 3	BD 21:15 DRIVE 3	BD SO 21:15 DRIVE 2	1:30 h	2 h	/db1/oracle /db2/oracle /exp/eflowme /opt	/usr /admin /patches
Myscdsc01 (5)	ORACLE-META4 (BD) SEMDMO1 (SO)	0	BD 21:30 DRIVE 4	BD 21:30 DRIVE 4	BD 21:30 DRIVE 4	BD 21:30 DRIVE 4	BD 21:30 DRIVE 4	BD 21:30 DRIVE 4	BD SO 21:30 DRIVE 4	2 h	2:40 h	/db1/oracle/tracta /db2/oracle/scripts /db3/oracle /db4/oracle /exp/eflowme /patches	/usr /opt /admin /patches
Myscdsc01 (6)	SEMAMPL01 (SO)	0							SO 21:30 DRIVE 5		1:30 h		/usr /opt /netbackup /admin /archive /vaz
Myscdsc01 (7)	ORACLE-APL01 (BD) SEMAMPL01 (SO)	0	BD 21:45 DRIVE 6	BD 21:45 DRIVE 6	BD 21:45 DRIVE 6	BD 21:45 DRIVE 6	BD 21:45 DRIVE 6	BD 21:45 DRIVE 6	BD SO 21:45 DRIVE 6	2 h	2:40 h	/db1/oracle /db2/oracle /db3/oracle /exp/eflowme	/usr /opt /admin /patches /vaz
Myscdsc01 (8)	ORACLE-APL02 (BD) SEMAMPL02 (SO)	0	BD 21:45 DRIVE 7	BD 21:45 DRIVE 7	BD 21:45 DRIVE 7	BD 21:45 DRIVE 7	BD 21:45 DRIVE 7	BD 21:45 DRIVE 7	BD SO 21:45 DRIVE 7	2 h	2:40 h	/db1/oracle /db2/oracle /db3/oracle /exp/eflowme	/usr /opt /admin /patches /vaz
Myscdsc01 (9)	ORACLE-MAXIMO (BD) SEMINT001 (SO)	0	BD 22:00 DRIVE 8	BD 22:00 DRIVE 8	BD 22:00 DRIVE 8	BD 22:00 DRIVE 8	BD 22:00 DRIVE 8	BD 22:00 DRIVE 8	BD SO 20:00 DRIVE 8	2 h	2:30 h	/db1/oracle /db2/oracle /db3/oracle /exp/eflowme	/usr /opt /admin /doad /vaz
Myscdsc01 (10)	ORACLE-BD01 (BD) SEMBD01 (SO)	0	BD 22:00 DRIVE 9	BD 22:00 DRIVE 9	BD 22:00 DRIVE 9	BD 22:00 DRIVE 9	BD 22:00 DRIVE 9	BD 22:00 DRIVE 9	BD SO 22:00 DRIVE 9	3:00 h	4:00 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /db5/oracle /exp/eflowme	/usr /opt /netbackup /admin /patches
Myscdsc01 (11)	ORACLE-CAL01 (BD) SEMICAL01 (SO)	0	BD 22:15 DRIVE 10	BD 22:15 DRIVE 10	BD 22:15 DRIVE 10	BD 22:15 DRIVE 10	BD 22:15 DRIVE 10	BD 22:15 DRIVE 10	BD SO 22:15 DRIVE 10	3:00 h	4:00 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /db5/oracle /exp/eflowme	/usr /opt /netbackup /admin /patches
Myscdsc01 (12)	ORACLE-BD02 (BD) SEMBD02 (SO)	0	BD 22:15 DRIVE 11	BD 22:15 DRIVE 11	BD 22:15 DRIVE 11	BD 22:15 DRIVE 11	BD 22:15 DRIVE 11	BD 22:15 DRIVE 11	BD SO 22:15 DRIVE 11	2:30 h	3:30 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /exp/eflowme	/usr /opt /admin /patches /vaz
Myscdsc01 (13)	ORACLE-APL03 (BD) SEMAMPL03 (SO)	0	BD 22:30 DRIVE 12	BD 22:30 DRIVE 12	BD 22:30 DRIVE 12	BD 22:30 DRIVE 12	BD 22:30 DRIVE 12	BD 22:30 DRIVE 12	BD SO 22:30 DRIVE 12	2:30 h	3:00 h	/db1/oracle /db2/oracle /db3/oracle /db4/oracle /exp/eflowme	/usr /opt /admin /patches /vaz
Myscdsc01 (14)	ORACLE-APL04 (BD) SEMAMPL04 (SO)	0	BD 22:30 DRIVE 13	BD 22:30 DRIVE 13	BD 22:30 DRIVE 13	BD 22:30 DRIVE 13	BD 22:30 DRIVE 13	BD 22:30 DRIVE 13	BD SO 22:30 DRIVE 13	2 h	2:40 h	/db1/oracle /db2/oracle /db3/oracle /exp/eflowme	/usr /opt /admin /patches /vaz

Fig. 4.23 Tabla de respaldos de la empresa.

Asignación lógica con la física

La empresa cuenta actualmente con: una unidad de almacenamiento Sun Storage L11000 y dos equipos Sun Enterprise 10000 con 14 dominios:

- 1- mxscbcon01: mx de México, scb de Servicios Corporativos Bimbo y con01 de Configuración 01.
- 2- mxscbdes01: mx de México, scb de Servicios Corporativos Bimbo y des01 de Desarrollo 01.
- 3- mxscbcap01: mx de México, scb de Servicios Corporativos Bimbo y cap01 de Capacitación 01.
- 4- mxscbdw01: mx de México, scb de Servicios Corporativos Bimbo y dw01 de Data Ware House01.
- 5- mxscbbdm01: mx de México, scb de Servicios Corporativos Bimbo y bdm01 de base de datos Mantenimiento 01.
- 6- mxscbaplm01: mx de México, scb de Servicios Corporativos Bimbo y apl01 de Aplicativo Meta 01.
- 7- mxscbapl01: mx de México, scb de Servicios Corporativos Bimbo y apl01 de Aplicativo 01.
- 8- mxscbapl02: mx de México, scb de Servicios Corporativos Bimbo y apl02 de Aplicativo 02.
- 9- mxscbmt01: mx de México, scb de Servicios Corporativos Bimbo y mto01 de Mantenimiento 01.
- 10- mxscbbd01: mx de México, scb de Servicios Corporativos Bimbo y bd01 de Base de Datos 01.
- 11- mxscbcal01: mx de México, scb de Servicios Corporativos Bimbo y cal01 de Calidad 01.
- 12- mxscbbd02: mx de México, scb de Servicios Corporativos Bimbo y bd02 de Base de Datos 02.
- 13- mxscbapl03: mx de México, scb de Servicios Corporativos Bimbo y apl03 de Aplicativo 03.
- 14- mxscbapl04: mx de México, scb de Servicios Corporativos Bimbo y apl04 de Aplicativo 04.

Todos los dominios excepto mxscbaplm01 tienen una base de datos, los dominios que contienen las bases de datos más importantes son: mxscbcon01, mxscbapl01, mxscbapl02, mxscbapl03, mxscbapl04, mxscbbd01 y mxscbbd02; al conjunto de estos dominios con sus bases de datos se les llama ambiente de producción, en ellos están las bases de datos, aplicaciones e interfaces que utilizan los usuarios desde las fábricas para llevar a cabo las labores diarias.

A los dominios restantes se les conoce como el ambiente de pruebas, en ellos se aplican parches, actualizaciones e instalación de software antes de ser aplicado en el ambiente de producción con la finalidad de reducir el riesgo de detener la producción a causa de alguna falla en la instalación.

Por lo general todos los dominios a nivel base de datos se respaldan una vez al día y el sistema operativo una vez a la semana. Dependiendo del número de pruebas y de actualizaciones a los ambientes pueden realizarse respaldos manuales adicionales solicitados por los usuarios.

El tamaño de las bases de datos es variable, la más grande de ellas -que está en el dominio mxscbcon01- tiene un tamaño de 12GB de información y 200GB de espacio disponible, mientras que la más pequeña es de 2GB de información y 30GB de espacio disponible.

Al dominio de mxscbcon01 se le asignaron 25 cintas de la librería L11000 para realizar el respaldo de información diario y semanal durante dos meses, a los dominios mxscbbd01 y mxscbbd02 se les asignaron 15 cintas para el respaldo diario y semanal de dos meses a cada uno, el dominio mxscbaplm01 tiene asignado 5 cintas para el respaldo semanal durante dos meses y los dominios restantes tiene asignados 10 cintas para el respaldo diario y semanal durante dos meses.

La asignación de las cintas se hizo de acuerdo al número que tienen asignados los dominios con respecto de la posición 0 del DLT de la librería, es decir:

- El dominio 1 tiene asignadas 25 cintas de la 0 a la 24
- El dominio 2 tiene asignadas 10 cintas de la 25 a la 34
- El dominio 3 tiene asignadas 10 cintas de la 35 a la 44
- El dominio 4 tiene asignadas 10 cintas de la 45 a la 54
- El dominio 5 tiene asignadas 10 cintas de la 55 a la 64
- El dominio 6 tiene asignadas 5 cintas de la 65 a la 69
- El dominio 7 tiene asignadas 10 cintas de la 70 a la 79
- El dominio 8 tiene asignadas 10 cintas de la 80 a la 89
- El dominio 9 tiene asignadas 10 cintas de la 90 a la 99
- El dominio 10 tiene asignadas 15 cintas de la 100 a la 114
- El dominio 11 tiene asignadas 10 cintas de la 115 a la 124
- El dominio 12 tiene asignadas 15 cintas de la 125 a la 139
- El dominio 13 tiene asignadas 10 cintas de la 140 a la 149
- El dominio 14 tiene asignadas 10 cintas de la 150 a la 159

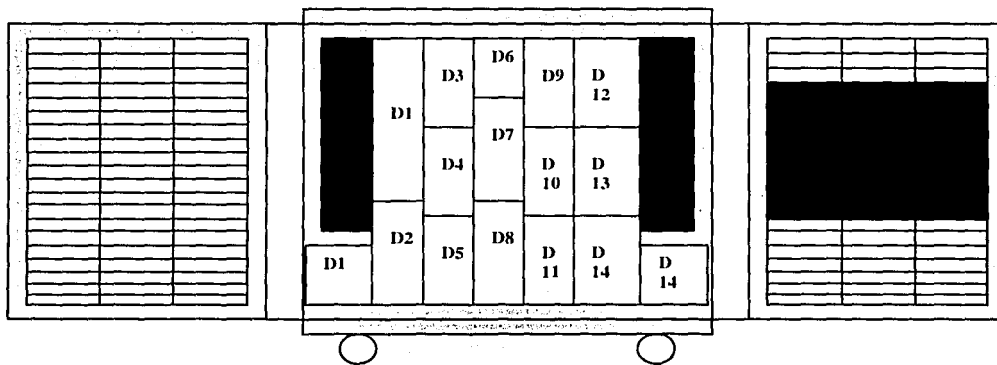


Fig. 4.24 DLTs asignados dentro de la librería con cuadros gris oscuro.

Existen otras cuarenta cintas que están designadas para diferentes respaldos, por lo general manuales y que no tienen día de expiración, es decir, no se reutilizarán, también en estas se incluyen tres cintas limpiadoras que serán utilizadas cuando así se requiera.

No existen respaldos históricos de información solo respaldos diarios y semanales, la información sólo podrá ser recuperada de dos meses atrás a la fecha, no es posible recobrar más atrás, excepto en casos muy especiales.

El número de cintas asignado dentro de la librería es de 200 y 126 disponibles para futuros respaldos o asignación a pools, en un inicio se tenía un número secuencial de asignación de cintas a cada dominio pero en la práctica se fue cambiando y ahora se asignan las cintas aleatoriamente.

Actualmente existen 126 cintas disponibles que se utilizan como pool schatch, es decir, están disponibles para cuando algún respaldo requiera una nueva cinta, pero, con esta cantidad es posible soportar otros cinco dominios asignandoles para sus backups 15 cintas a cada uno y aún sobrarían 41 cintas como reserva.

A continuación se muestra una gráfica de la duración de los respaldos:

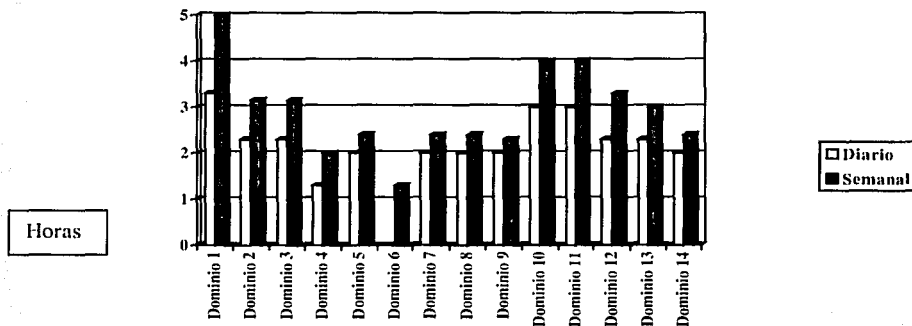


Fig. 4.25 Duración de los respaldos diarios y semanales.

En la gráfica anterior puede verse que el respaldo diario y semanal del dominio unos duran más tiempo que los otros y el respaldo diario del dominio 4 es el que dura menos tiempo.

Supongamos que en cierto momento debemos respaldar todos los dominios para poder realizar un mantenimiento general a las máquinas, el tiempo que necesitaríamos sería de 10 horas. En un día de respaldo semanal normal todo el proceso de respaldo se lleva cinco horas pero previniendo cualquier contratiempo se toman otras cinco horas más.

El tiempo promedio de duración aproximado de los respaldos diarios es de 2 horas con 10 minutos y el tiempo promedio aproximado de los respaldos semanales es de 3 horas.

La siguiente figura presenta los drives de la librería en donde se realiza el respaldo de los dominios, aunque estos pueden ser asignados o reasignados según las necesidades.

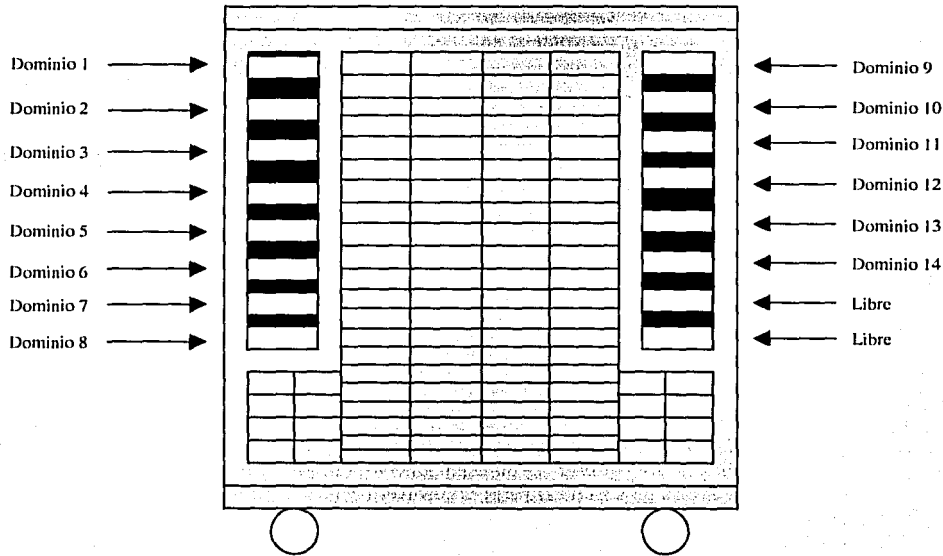


Fig. 4.26 Asignación de los drives para el respaldo de los dominios.

Dentro de la librería quedan dos drives disponibles que, como se mencionó, se utilizan caso de emergencia, pero con la cantidad de cintas disponibles que se encuentran dentro de la librería y con los dos drives disponibles, fácilmente se podrían agregar cinco dominios más que podrían ser respaldados diariamente sin problemas.

Con los equipos que se tiene actualmente y el esquema de trabajo podría trabajarse así durante tres años como mínimo, existen bastantes cintas disponibles dentro de la librería, dos drives libres en la librería y la posibilidad de lanzar respaldos en los mismo drives cuando termine el backup en proceso, y las bases de datos fueron diseñadas por los DBA con vista a un crecimiento soportado de tres años.

RESUMEN CAPÍTULO IV

Una función primordial en la administración del sistema operativo Solaris es el manejo de los backup, que son herramientas de manejo y almacenamiento de información como medidas de seguridad. Existen varios dispositivos para almacenar los datos del sistema, cada uno con funciones y características específicas, entre ellos se encuentran los discos flexibles (disquettes), disco duro —o una partición de éste—, las cintas magnéticas QIC y DAT, los CD-ROM, y los cartuchos DLT.

La importancia de las copias de seguridad (backup) radica en que son una parte imprescindible en la planificación de la seguridad de todo el sistema, los procedimientos para elaborar copias de seguridad y restauración tienen gran trascendencia, pues previenen la pérdida de datos valiosos en la computadora. Un *backup* se puede definir como el proceso de copiar datos del sistema o información almacenada permanentemente en la computadora, a otro medio, mientras que *restaurar* se refiere al procedimiento de devolver al sistema versiones o sistemas de archivos o información con base en la copia de seguridad.

Una correcta copia de seguridad garantiza la recuperación de una versión anterior de la información que pudiera haberse borrado o modificado, de igual modo, permite recuperar los archivos de configuración del sistema en caso de resultar dañados o ser destruidos.

Existen algunos elementos a considerar para garantizar que los procedimientos de respaldo cumplan sus objetivos, como verificar el correcto funcionamiento de los backups, seguir ciertas recomendaciones al momento de etiquetar los dispositivos de almacenamiento y seleccionar un lugar adecuado para almacenar las copias de seguridad.

Las copias de seguridad se manejan de acuerdo a las necesidades del administrador, el equipo, la empresa y la información almacenada, pues los dispositivos de almacenamiento varían según sus características específicas. El respaldo de la información en copias de seguridad permite resolver problemas originados por errores en el manejo de datos, pudiendo restaurar información anterior.

Existen diferentes formas, funciones y equipos de backup en el mercado, uno de ellos es el equipo Sun StorEdge Enterprise NetBackup, que es un sistema de respaldo, archivado y servicios de restauración para clientes y servidores en redes cliente/servidor, cuya principal ventaja es proporcionar un alto rendimiento, además de que su rango de operación puede hacerse desde un sistema standalone a uno entire enterprise.

Los dispositivos de almacenamiento de NetBackup pueden ser discos, cintas u ópticos, además, está diseñado para que otros dispositivos secundarios de almacenamiento puedan ser compartidos por otros productos Sun StorEdge. El administrador de NetBackup puede calendarizar los respaldos, atender operaciones de respaldo por red, y respaldar sólo archivos recientes, de modo que archive sistemáticamente los respaldos con base en un tiempo determinado, agilizando este procedimiento sin interferir en la labor de la empresa. Estos respaldos pueden ser manuales o automáticos, el primer caso se emplea solamente en circunstancias extraordinarias.

Netbackup es de fácil manejo, sus usuarios pueden realizar las tareas de archivar, respaldar y recuperar información sin ocupar al administrador, además, mantiene una base de datos en el servidor con la información de los respaldos y las restauraciones.

Una de las principales ventajas de Netbackup cluster es su alta disponibilidad que asegura operaciones continuas en el medio ambiente maestro/esclavo y permite que los datos en el cluster sean protegidos y recuperados rápidamente en caso de cualquier eventualidad.

Por otro lado, existe un equipo automatizado de respaldo y recuperación: la librería Sun StorEdge L11000. Algunas de sus ventajas son: gran capacidad, alto rendimiento de respaldo y recuperación, se pueden unir hasta cinco librerías en una unidad virtual, tiene una compatibilidad garantizada con futuras tecnologías, está protegida contra fallas en las fuentes de poder y sobrecalentamiento, es de fácil manejo, puede ser reparada sin necesidad de estar apagada, tiene fácil acceso para reemplazar componentes críticos, tiene un amplio rango de configuraciones relacionadas con servicio, entre otras.

Con el objetivo de visualizar los elementos de la librería Sun StorEdge L11000, se presentó un ejemplo real de una empresa donde se maneja el sistema operativo Solaris 2.6, y el software NetBackup 3.2 para llevar a cabo los respaldos de información; dicha empresa cuenta con tres diferentes dominios (mxscbcon01, mxscbcal01 y mxscbbd01) con Solaris 2.6, una librería L11000, clientes en dos Sun Ultra Enterprise 10000 con 14 dominios y una red de administración y respaldos.

Una vez descritas las características del servidor maestro (mxscbcon01), de los servidores esclavos (mxscbcal01 y mxscbbd01) y de la librería L11000 se describió el proceso de inicio de NetBackup 3.2, monitoreo de dichos procesos y su finalización con el comando; también se describieron los comandos necesarios para iniciar y terminar los demonios al tiempo del boot del servidor de respaldos al momento de hacer shutdown. A continuación se presentó el comando #xnb & para desplegar el ambiente gráfico de NetBackup con su menú de cinco opciones: xbp, xbpmon, xbpadm, xvmadm, y xdevadm.

Se vió el proceso de crear clases como parte imprescindible de la configuración de una política de respaldo, definiendo la clase como el conjunto de información para realizar respaldos automáticamente; el procedimiento para crear una nueva clase es mediante la ventana del administrado xbpadm, que permite definir sus diferentes partes: atributos, clientes, schedules y files. Posteriormente, se inicia un respaldo manual o calendarizado de acuerdo a la definición del horario y las necesidades de la información.

Finalmente, se presentó un ejemplo de una política de respaldo real, aclarando que los procedimientos de respaldo dependerán del sistema que se está administrando y de las necesidades particulares de cada empresa.

CONCLUSIONES

CONCLUSIONES

Solaris es uno de los sistemas operativos más importantes en la actualidad, sus características proporcionan un sin número de ventajas para manejar grandes cantidades de información, bases de datos, manejo de redes y estaciones de trabajo, etcétera, además, su funcionamiento garantiza un mínimo de complicaciones y fallos.

A esto se debe el gran auge que está tomando UNIX Solaris en el mercado informático, pues cada vez más las grandes empresas están migrando su información a estos sistemas, ya que ofrecen más alternativas y ventajas que otros.

Consecuentemente, el mercado laboral requiere cada vez más administradores UNIX Solaris, que cuenten con los conocimientos básicos sobre estos sistemas para administrar diferentes equipos y manejar grandes cantidades de información.

Por ello es importante que los estudiantes o recién egresados de la carrera de Ingeniería en Computación (ICO), cuenten con los conocimientos básicos sobre este sistema, de modo que sean más competitivos en el mercado laboral y cuenten con mayores posibilidades de desarrollo profesional.

Si bien es cierto que todas las ramas de la informática ofrecen posibilidades de empleo y desarrollo, en la actualidad, por la saturación de técnicos y profesionistas en computación, las mejores posibilidades de superación se concentran en quien conozca más sobre los diferentes sistemas operativos y los diferentes equipos informáticos.

En este sentido, es prioritario de los estudiantes de ICO egresen de la carrera con conocimientos básicos sobre los mismos para poder abrirse paso en el mundo laboral. Lamentablemente, no existe en nuestra escuela documentación sobre este sistema operativo en particular, ni cursos o asignaturas que lo contemplen, lo que provoca que los alumnos y egresados lo desconozcan casi totalmente y les sea más difícil incorporarse a trabajos relacionados con él.

Uno de los principales problemas asociados al conocimiento de Solaris es la falta de material informativo sobre él, pues aunque se pueden encontrar una gran variedad de libros sobre sistemas operativos, éstos por lo general abordan los temas superficialmente.

Por otro lado existen manuales de usuario sobre este sistema operativo, pero es difícil tener acceso a ellos, pues se venden casi exclusivamente con equipos nuevos, o sólo se otorgan si se ha tomado un curso sobre este sistema, obviamente, ambas alternativas son muy costosas y no están al alcance de todos los egresados de ICO.

Eventualmente, estos manuales pueden comprarse independientemente, pero también así tienen un costo elevado y la mayoría contienen la información en inglés, lo que representa otra desventaja para el estudiante o recién egresado de la carrera.

Por otra parte, Sun Microsystem cuenta con una página en Internet que contiene una enorme cantidad de información sobre sus equipos y sistemas, sin embargo, la mayoría del sitio está en inglés, y es tan grande y complejo que se requiere mucho tiempo y recursos para localizar información práctica e imprimirla.

Otra opción para tener contacto con información sobre el sistema Solaris, es cuando se trabaja en una empresa que cuenta con este sistema, sin embargo, es difícil que alguien sin conocimiento previo sobre él pueda acceder a operarlo, además, la información con la que cuentan los administradores de este sistema es monopolizada y restringida para un nuevo empleado, sobre todo sino demuestra tener nociones sobre este sistema.

Aparte de los manuales de Sun o la página de Internet no existe información documental en México sobre el sistema operativo Solaris. De igual manera, la información en español disponible en Internet sobre este sistema, generalmente es muy pequeña y dispersa, dejando en el lector grandes huecos de información y dudas sin resolver, pues presentan información teórica de algunos problemas comunes con estos sistemas, pero no ofrecen ejemplos sobre cómo resolverlos.

Una última alternativa para conocer Solaris es tomar cursos especializados de administración, aunque generalmente, los buenos cursos son demasiado caros y los económicos no cubren con la cantidad y calidad de información necesaria para que el participante pueda administrar este sistema.

Por lo anterior, este trabajo puede servir como guía básica de Solaris para los estudiantes y egresados de ICO, de modo que les sirva para tener las nociones básicas sobre este sistema cuando se incorporen al mercado laboral. Asimismo, este texto puede servir si se va a tomar un curso de Solaris, pues permitiría tener nociones sobre los temas presentados y aprovechar mejor la capacitación.

Igualmente, la información que se ha presentado en este trabajo, puede servir para otros sistemas y equipos, pues aunque los fundamentos y conceptos expuestos se basan en el sistema Solaris 2.7, éstos se pueden aplicar a futuras versiones o equipos.

Al momento de realizar esta investigación, la versión 2.7 de Solaris era la más actual; dada la vertiginosidad de los avances tecnológicos, es muy probable que muy pronto salga al mercado la versión 2.8 de este sistema. Empero, los conceptos que he presentado en este trabajo pueden servir igualmente para nuevas versiones, bastará con familiarizarse con la nueva versión y localizar la analogía correspondiente en los comandos y funciones.

El sistema Solaris contiene una gran cantidad de aplicaciones y características, por ello, cada uno de los temas que se han abordado en este trabajo puede ser retomado por alguien más e investigarse profundamente, pues por cada uno se puede realizar una extensa investigación o, incluso, un trabajo de tesis particular.

Este trabajo está pensado para que los nuevos egresados de ICO cuenten con un material básico que conjugue los conocimientos recién adquiridos en la escuela, con los elementos que se tendrán que desarrollar en el mercado laboral; es una aportación a los futuros administradores del sistema operativo Solaris, y a la ENEP Aragón, pues pretende ser un material de apoyo para su alumnado, de modo que complete su formación académica y le abra sus expectativas.

Sin embargo, a lo largo de mi investigación pude percatarme que un administrador de sistemas no se forma sólo con los conocimientos académicos, teóricos y prácticos sobre determinado sistema operativo, sino que estos conocimientos deben conjugarse con su experiencia en el manejo del mismo, su interés por superarse y el empeño en desarrollar correctamente su profesión. Es decir, un administrador del sistema operativo se forma con la teoría y la práctica, con la retroalimentación entre documentación y experiencia.

De igual manera, pude darme cuenta de la importancia de conocer el inglés técnico, pues esto facilitaría enormemente la comprensión del funcionamiento de los sistemas operativos, los equipos y su documentación en general.

Con esta investigación me he percatado de que egresar de la carrera no es el término de un proceso de aprendizaje, tan sólo es el principio de una etapa de formación profesional, personal y laboral, en la que se aplicarán los conocimientos adquiridos en la escuela, por ello, qué mejor, que el alumnado de la carrera cuente con más material informativo sobre los diferentes sistemas operativos para incorporarse al mercado laboral.

En este sentido, este material es una aportación para el alumnado de ICO de la ENEP Aragón, a modo de un manual de ejemplos claros y prácticos con la información más sustancial sobre la administración de Solaris, los temas que en él se desarrollaron son los fundamentos que todo administrador de sistemas debe poseer, a fin de que esta información pueda contribuir a su formación profesional.

TESIS DE GRADO
FALLA DE SISTEMAS OPERATIVOS

221

BIBLIOGRAFÍA

BIBLIOGRAFÍA

- Antonio Villalón Huerta. *Seguridad en UNIX y Redes Versión 1.2*. 2 de Octubre de 2000. España.
- Budgen, D. *Introducción al sistema operativo UNIX*. Ed. Gustavo Gill, España, 1987. 236p.
- Centro de Especialización SILVATECH *Solaris 2.X Network Administration* México D.F., Noviembre 1993.
- E. Alcalde, J. Morena, J. A. Pérez Campanero. *Introducción a los Sistemas Operativos*. Mc Graw Hill, 1992
- Juanes Baza, David. *Sistemas Operativos*. Paraninfo, España, 1991. 367p.
- Milenković, Milán. *Sistemas Operativos. Conceptos y Diseños*. Mc Graw Hill, España, 1998. 618p.
- Rachel Morgan y Henry Mc Gilton. *Introducción al UNIX Sistema V*. Mc Graw Hill 1990.
- Rosen, Kenneth, et al. *UNIX. Sistema V. Versión 4*. Traducción: Daniel Mozos Muñoz. Mc Graw Hill, España, 1997. 1337p.
- Stalling, William. *Sistemas Operativos*. Noriega Editores, México, 1995. 845p.
- Sun Educational Services. *Solaris 7 System. Administration I*. Sun Microsystems, California, 1998 Tomo I y II.
- Sun Educational Services. *Solaris 7 System. Administration II*. Sun Microsystems 1998. Tomo I y II.
- Sun Educational Services. *Fundamentals of Solaris 2.x*. Sun Microsystems, California, 1999. Tomo I y II
- Tanenbaum, Adrew. *Sistemas Operativos. Diseño e Implementación*. Prentice-may Hispanoamericana, México, 1998. 741p.
- Tapia Martínez, Jesús Alejandro. *Administración del ambiente operativo UNIX como sistema abierto*. Tesis de Ingeniería en Computación, ENEP Aragón, UNAM, México, 1995. 118p.

PÁGINAS DE INTERNET

- <http://www.docs.sun.com>
- <http://www.cybercursos.net>
- <http://www.sun.com>
- <http://www.sunsite.unam.mx>
- <http://www.sis.ucm.es/SIS/UNIX/indice.htm>
- <http://www.kriptopolis.com>
- <http://www.geocities.com/petsburgh/haven/2095/index.html>
- <http://www.openbsd.org.mx/>
- <http://www.carsoft.com.ar/permisos.htm>
- <http://www.unixgroup.com>
- <http://lem.cui.upm.es>
- <http://members.es.tripod.de/zmanuel>
- <http://gencura.ugr.es/~jmerelo/ic/node1.html>
- <http://lucas.simplenet.com/trabajos/unix/unix.html>
- <http://www.geocities.com/SiliconValley/Lab/2641/unix.html>
- <http://www.geocities.com/petsburgh/haven/2095/first.html>
- <http://www.sun.com/mexico/>