

**Facultad de Ciencias Exactas y Naturales
- Universidad de Buenos Aires -**

**BCR un sistema de Backup y Restore en caso de
Catástrofe**

Director: Lic. Roberto Bevilacqua

**Tesis de Licenciatura
Lic. en Ciencias de la Computación**

Mauro Castagna	88/87
Carlos Paz Soldán	690/87

Agradecimientos

Al Lic. Roberto Bevilacqua, por su excelente predisposición y sus invaluables e instructivos comentarios.

A nuestros padres, que gracias a su sacrificio nos permitieron alcanzar este objetivo.

A nuestras esposas, Maria Fernanda e Irma, por su paciencia, compañía y apoyo.

A todos los que nos apoyaron y acompañaron.

INDICE

ABSTRACT	3
RESUMEN	4
ESTADO ACTUAL	5
OBJETIVOS	6
ENTORNO DE TRABAJO	6
1. PROTECCIÓN DE LA INFORMACIÓN	6
1.1 PROTECCIÓN DE LA INFORMACIÓN POR HARDWARE Y/O SOFTWARE.....	6
Disk Array	6
Nivel 0: Stripe Sets	7
Nivel 1: Mirror Sets	7
Nivel 5: Stripe Sets with Parity	8
1.2 PROTECCIÓN DE LA INFORMACIÓN SÓLO POR SOFTWARE.....	8
1.2.1 Backup OnLine	8
Backup Completo.....	9
Backup Incremental.....	9
Backup Diferencial.....	9
1.2.2 Backup Offline	9
2. ARQUITECTURA Y TECNOLOGÍA	10
2.1 ARQUITECTURA DE UN DISCO RÍGIDO.....	10
2.2 TIPOS DE BUS.....	10
• Industry Standard Architecture – ISA	10
• Extended ISA – EISA	10
• Peripheral Component Interconnect – PCI	11
2.3 TIPOS DE CONTROLADORES.....	11
• ST506	11
• Enhanced Small Devices Interface – ESDI	11
• Integrated Drive Electronics – IDE	11
• Small Computer System Interface – SCSI	11
2.4 BIOS.....	12
• Por qué el BIOS es importante?	12
• Arquitectura del BIOS	12
• El test	13
3. ACCESOS A DISPOSITIVOS	14
3.1 CODIFICACIÓN DE DATOS.....	14
• Codificación FM (Modulación de Frecuencia)	14
• Codificación MFM (Modulación de Frecuencia Modificada)	14
• Codificación RLL (Codificación de Longitud Limitada)	15
3.2 ACCESO A DISCOS RÍGIDOS POR MEDIO DEL BIOS.....	15
3.3 ACCESO A DISCOS RÍGIDOS CONECTADOS A CONTROLADORAS.....	16
• ST506	17
• ESDI	18
• IDE	18
• SCSI	18
3.4 OPTIMIZACIÓN DE LA BÚSQUEDA EN UN DISCO RÍGIDO.....	19
Planificación FCFS (primero en llegar, primero en ser servido)	19
Planificación SSTF (menor tiempo de búsqueda primero)	19

<i>Planificación SCAN</i>	20
<i>Planificación SCAN de n-pasos</i>	21
<i>Planificación C-SCAN</i>	21
<i>Esquema Eschenbach</i>	22
<i>Optimización Rotacional</i>	22
4. SISTEMAS OPERATIVOS	22
<u>TRABAJO REALIZADO:</u>	25
<u>"BCR UN SISTEMA DE BACKUP Y RESTORE EN CASO DE CATASTROFE"</u>	25
1. ESQUEMA GENERAL DEL BCR	26
<i>Configuración de dispositivos</i>	26
<i>Realización de backup</i>	27
<i>Realización de restore</i>	28
2. CONSIDERACIONES PARA OBTENER UNA RESTAURACIÓN CONSISTENTE	29
3. EJEMPLO DE UN ESCENARIO EN CASO DE CATÁSTROFE	29
4. DESCRIPCIÓN DEL BCR	30
5. CONCLUSIONES	32
6. TRABAJOS A FUTURO (PROYECCIONES)	33
<u>ANEXOS</u>	34
1. ANÁLISIS FUNCIONAL DEL BCR	34
2. DESCRIPCIÓN DE PROCESOS	37
1) Establecer Configuración	37
2) BackUp	37
3) Restore	38
3. ESCENARIOS DE PRUEBAS	38
4. INSTALACIÓN Y MANUAL DEL USUARIO	40
5. BIBLIOGRAFÍA	41
6. CÓDIGO FUENTE	42

ABSTRACT

The backup of information is one of the most principal points that an organization is willing to solve. Another point to be taken into account is the complexity with which the said information can be recovered. It is necessary that the start up of the recovering process be less complex to secure the recovery of information, consistency, and, that the organizations are operative.

This work is strongly centralized on the total recovery, information and operating system; therefore, if any disaster occurs, the information can be recovered to its consistent state and with less complexity for the operators. In order to achieve these results, we have studied the behavior of three great components and its interactions:

Operating System:

The objective of the study of operating system was to develop a software that is independent from the operating system with which the equipment to be recovered operates.

In order to perform the disk restoration containing the information, it is necessary to investigate the handling of file systems and drivers used by the operating systems.

Hardware:

A research was carried out regarding the different controller cards, IDE, SCSI interphases, the storage mode of information in hard disks as well as the manner to access to them.

BIOS:

The objective of BIOS investigation is to use it as a gateway for the access to hardware.

At the end of the study of this three components. We design a software that allow us to do a backup and restore separately from the operating System installed in the server. Since, the cost of testing the different hardware configurations is currently high; Our proposal is to focus on Intel platform for the easy hardware access and software that works on this architecture.

RESUMEN

El resguardo de la información es uno de los puntos principales que una organización se preocupa por solucionar. Otro de los puntos que se tiene en cuenta es la complejidad con la que se puede poner en marcha la recuperación de dicha información, lo deseable es que sea lo menos complejo posible, de tal manera de asegurar que la información sea recuperada y mantenga su consistencia y que la organización siga operativa.

Este trabajo está fuertemente enfocado al resguardo total de la información, sistema operativo e información almacenada sobre los discos, para que en caso de catástrofe se pueda recuperar la información a un estado **consistente** y con **poca complejidad** para los operadores. Para lograr esto estudiamos el comportamiento de tres grandes componentes y sus interacciones:

Sistema Operativo:

El objetivo del estudio de los sistemas operativos fue realizar un software que sea independiente del sistema operativo con el cual opera el equipo a ser resguardado. Realizar la restauración de discos que contengan información, para ello es necesario investigar el manejo de los file systems y drivers que utilizan los sistemas operativos.

Hardware:

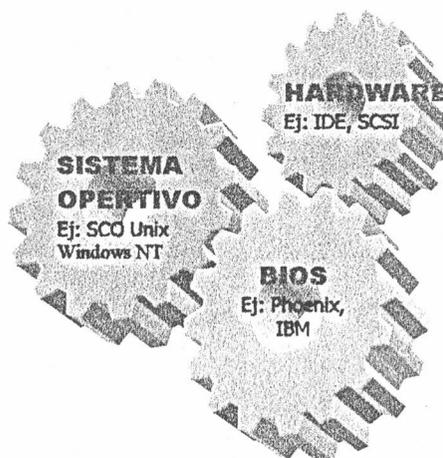
Se investigó las distintas controladoras, sus interfaces (IDE, SCSI) y modo de almacenamiento de información en los discos rígidos y la manera de acceso a los mismos.

BIOS:

El objetivo de la investigación del BIOS fue el de utilizarlo como un gateway para el acceso al hardware o discos rígidos.

Al finalizar el estudio de estos tres componentes diseñamos un software que nos permite realizar un backup y un restore independientemente del sistema operativo que tenga instalado el servidor.

Debido al costo que implica probar las distintas configuraciones de hardware existentes en la actualidad nuestro trabajo se centró sobre la plataforma Intel por el fácil acceso al hardware como así también al software que funciona sobre esta arquitectura.



ESTADO ACTUAL

La pérdida de información en sistemas de computación es siempre un punto crítico que causa verdaderas catástrofes en miles de organizaciones.

Podemos mencionar algunas causas por la cual se puede perder información:

- Virus.
- Cortes de energía.
- Ruptura de disco.
- Errores involuntarios de operadores.
- Falla del sistema operativo.
- etc.

Algunas de estas causas son relativamente nuevas, como son los virus informáticos, algunas de ellas se han podido controlar, aunque sea en forma parcial, como los cortes de energía, el mal funcionamiento del hardware, etc., pero todos los sistemas informáticos están expuestos a la posibilidad de pérdida de información tanto sea por problemas de hardware como de software.

Distintos tipos de pérdida de información:

- 1) Pérdida de archivos de datos, inconsistencia de datos.
- 2) Pérdida de archivos o configuración del sistema operativo.
- 3) Ruptura del Hardware de almacenamiento de la información.

Actualmente los sistemas de backup de plataforma Intel están diseñados para resguardar la información a nivel de archivos y sistema de directorio, lo cual soluciona los tipos de pérdida descritos en el punto 1) y dependiendo del sistema operativo en uso y el software de backup utilizado, se resuelven los puntos 2), 3) con mayor complejidad y la mayoría de las veces por personal especializado.

La tendencia actual de las empresas es tener la menor cantidad de personal en el departamento de Sistemas o contar con un apoyo externo especializado, con lo cual hay que tener el mayor resguardo de la información tanto de datos, como del sistema operativo, para poder realizar una recuperación de esto en el menor tiempo posible y de la forma más simple y segura.

Los productos de hoy en día para plataformas Intel que realizan backups son dependientes del sistema operativo y orientados a la recuperación de archivos, por lo tanto requiere en caso de catástrofe realizar la instalación parcial o total del sistema operativo y tener un mayor conocimiento de los productos que tenemos instalados en nuestra organización para poder tener un buen respaldo.

OBJETIVOS

Los objetivos que nos propusimos que cumpliera nuestro trabajo son los siguientes:

- Diseñar un mecanismo de backup y restore que sea independiente del sistema operativo en uso, la plataforma utilizada y la información almacenada.
- Diseñar un mecanismo de backup y restore que permita en caso de "catástrofe" volver a un estado operable, seguro y consistente.
- Diseñar un mecanismo de backup y restore que requiera bajos conocimientos por parte del operador.

ENTORNO DE TRABAJO

Dada la existencia de una gran variedad de hardware de distintos fabricantes y el acceso a esos productos es una inversión de dinero, el cual no disponemos, definimos el entorno de nuestro trabajo sobre la plataforma Intel, controladoras cuyo interfase es IDE y SCSI; sin embargo, dejamos la posibilidad de que nuestro trabajo pueda ser aplicado a otras tecnologías a las cuales no hemos tenido acceso y que el diseño del mismo sea fácilmente aplicable.

A continuación detallaremos la tecnología que investigamos para el desarrollo de nuestro trabajo.

1. Protección de la información

1.1 Protección de la información por Hardware y/o Software

Disk Array

Un array de discos consiste en múltiples discos rígidos coordinados por una controladora. La información almacenada se encuentra distribuida entre los discos que conforman el array. Por ejemplo un archivo puede estar distribuido en más de un disco. Dependiendo de que nivel de RAID sea utilizado se puede obtener una mejor performance y/o seguridad.

No obstante "*no hay tolerancia a falla hasta que la falla es reparada*" y muy pocas implementaciones de RAID pueden soportar dos fallas simultáneas.

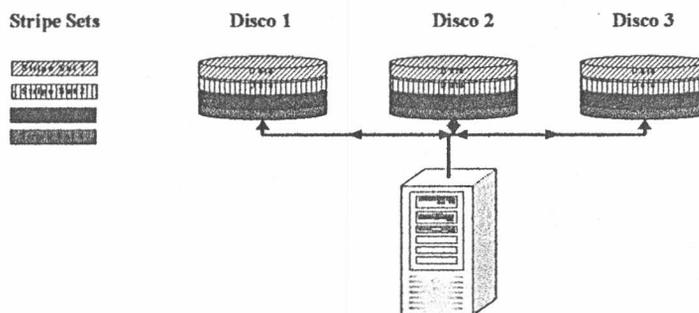
Si se tiene implementado RAID de nivel 5 y uno de los discos falla y es reemplazado, los datos pueden ser regenerados sin tener que restaurar la información de algún backup y aplicar las últimas transacciones producidas para obtener la información en el estado anterior al de producirse la falla, esta es una de las ventajas que brinda el Disk Array el de tener una alta disponibilidad de los datos.

A continuación mencionaremos los niveles más utilizados:

Nivel 0: Stripe Sets

Los Stripe sets están conformados por áreas de espacios libres de distintos discos que se convierte en un solo volumen lógico. Los datos están esparcidos entre los diferentes discos que conforman el array.

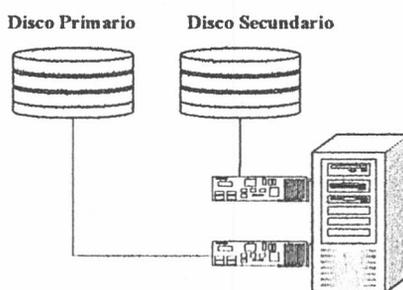
El Stripe Set no provee ningún tipo de tolerancia a falla si bien mejora el performance de acceso a la información.



Nivel 1: Mirror Sets

El espejado de discos o Mirror Sets, es donde dos drives almacenan idéntica información, toda escritura es realizada sobre los dos discos, el primario y el disco espejo, con lo cual la performance de escritura se degrada; para evitar esto muchos implementan disk duplexing, donde cada drive tiene su propia controladora.

Si en el Mirror Sets uno de los dos discos falla el sistema utiliza la información del otro disco. Esta solución tiene un alto costo para implementar por que sólo se utiliza el 50 % de todo el espacio que brindan los discos que componen el mirror sets. Cuando se utiliza mirror sets se recomienda la utilización de discos del mismos tamaño, modelo y fabricante.

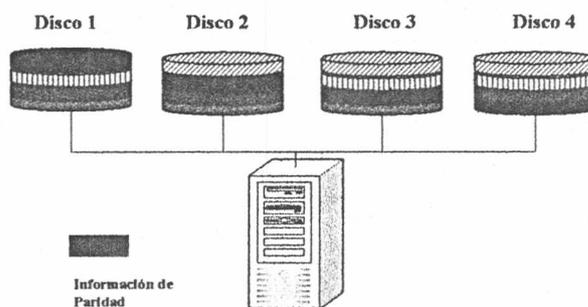


Mirror Sets tiene mejor performance de escritura y lectura que el RAID de Nivel 5, además cuando se pierde alguno de los discos miembros del Mirror Set la performance del sistema no disminuye. Si bien es más caro por el espacio que no utiliza, se puede implementar con sólo dos discos y lo aconsejable es en controladoras diferentes.

Nivel 5: Stripe Sets with Parity

El Nivel 5 es conocido como striping with parity. Los datos están distribuidos en bloques a través de todos los discos. El dato de redundancia es provisto por la información de paridad.

El dato y la información de paridad son distribuidos en el array pero siempre en discos diferentes.



Strip sets with parity tiene un mejor performance que Mirror Sets (Nivel 1). Sin embargo cuando un disco miembro del array falla la performance de lectura y escritura se degrada por la necesidad de reconstruir con la información de paridad el dato solicitado. Esta solución es recomendada cuando las aplicaciones requieren redundancia y son primordialmente orientadas a lectura, en esta solución la performance de escritura se reduce por el cálculo de paridad.

1.2 Protección de la información sólo por Software

La forma de realizar backups de los productos de software (S.O., Bases de Datos, etc.) podríamos clasificarlos en dos:

- Backup Online
- Backup Offline

1.2.1 Backup OnLine

Decimos que un software realiza un backup OnLine cuando el acceso a los datos es permitido a medida que el backup se va realizando. El software se encarga de llevar el control de la consistencia de la Información monitoreando el acceso de los usuarios a la información. La mayoría de los productos de software que realizan backups lo hacen a nivel archivos, si algún usuario esta utilizando archivos en el momento de realizar el backup no se realiza el resguardo de los mismos, no así los productos de software que realizan backups de bases de datos que suelen realizar backups OnLine asegurando la consistencia de los datos almacenados en la base de datos, aunque es siempre recomendable la no utilización de la misma.

A continuación describiremos como ejemplo cómo realiza el backup OnLine MS SQL Server (RDBMS): MS SQL Server saca una foto del estado actual de la base de datos a la cual le realizará el backup.

A medida que realiza el backup página a página que es la unidad mínima de la base de datos, en cuanto un usuario quiere actualizar una página que todavía no fue resguardada SQL Server salta a esa página, realiza el backup y libera dicha página para el acceso del usuario, de esta manera SQL Server asegura la consistencia e integridad de los datos que son resguardados, y permite tener un backup On-Line. En la figura 1.2 mostramos la secuencia en la que realizaría el backup. [Ad98]

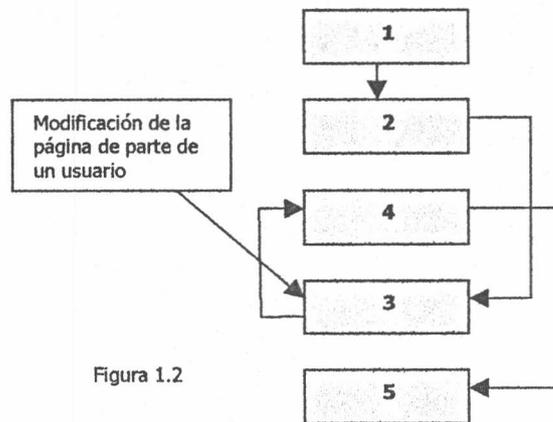


Figura 1.2

Los backup On line en su mayoría permiten realizar el respaldo en las siguientes modalidades:

Backup Completo

Un backup es completo cuando se respalda toda la información seleccionada. En casos en que se guardara registros de modificaciones, después de realizar el backup completo mayormente los mismos son eliminados, determinando un nuevo punto inicial de registración dado que se asume que tiene un respaldo de toda la información.

Backup Incremental

La Información que se resguarda es la información modificada desde el último backup incremental o backup completo. Para realizar la restauración de la información es necesario empezar del backup completo e ir restaurando los backups incrementales sin perder la secuencia de los mismos.

Backup Diferencial

El backup diferencial realiza el respaldo de todos los cambios de la información desde el último backup completo. Para la restauración de la información sólo es necesario el backup completo y el último backup diferencial.

1.2.2 Backup Offline

Decimos que un software realiza un backup Offline cuando los usuarios no pueden utilizar la información que va a ser resguardada con el software hasta la finalización de la misma, la mayoría de los softwares ponen en modo exclusivo la información de la cual se realizará el backup, de esta manera impide el acceso a la información hasta que el backup se realice por completo.

2. Arquitectura y Tecnología

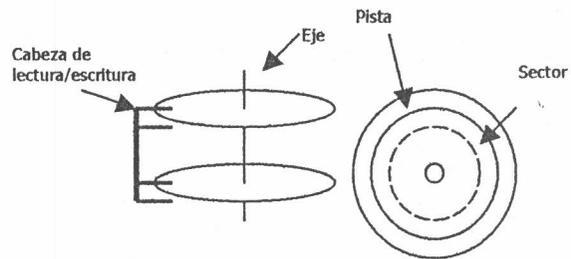
2.1 Arquitectura de un disco rígido

Las unidades de disco y el sistema operativo determinan la capacidad de los discos utilizados, pero la estructura del disco es esencialmente la misma, independientemente de la organización.

En la figura representamos una organización física y simplificada de un disco magnético.

El medio para el almacenamiento de datos es la capa de óxido magnético que cubre un disco, cuyo aspecto es parecido al de un disco fonográfico. Una unidad de disco puede disponer de uno o más de estos discos para escribir y leer datos.

A diferencia de las cintas los discos giran constantemente.



Los datos se leen y se escriben por medio de las cabezas de lectura/escritura montadas en un conjunto de cabezas, de tal manera que puedan entrar en contacto con la porción del disco donde residen los datos requeridos.

Los datos se almacenan en la superficie del disco magnético en círculos concéntricos llamados pistas, la colección de pistas de todas las superficies que se encuentran a la misma distancia del eje del disco se llama cilindro. El número de pistas en una sola superficie determina el número total de cilindros. Cada pista se divide posteriormente en segmentos, llamados sectores. La cantidad de datos que puede ser almacenada en cada cara de un disco depende del número de pistas (su densidad) y el tamaño de sus sectores. [Mi88]

2.2 Tipos de Bus

Un bus es un componente de las PC al cual se le conecta dispositivos. A continuación mencionaremos los distintos tipos de bus que existen en el mercado:

- **Industry Standard Architecture – ISA**

Industry Standard Architecture. Primeramente existió de 8-bit y posteriormente de 16-bit provee la interface entre los dispositivos o placas con el bus interno de una PC. [Pe85]

- **Extended ISA – EISA**

Los dispositivos EISA existentes proveen un mecanismo estándar de identificación y método para configurar sus dispositivos a través de software, teniendo como estándar 32-bit de transmisión pero manteniendo la compatibilidad con ISA. [Pe85]

• **Peripheral Component Interconnect – PCI**

Arquitectura que transmite simultáneamente 32 bits que corre normalmente a 33 Mhz a través de una conexión de 124 pines, como también logra realizar una transmisión de 64 bits pero a través de una conexión de 188 pines a una velocidad de 66 Mhz.

[Pe85]

2.3 Tipos de controladores

Los discos rígidos, al igual que los floppy disk, requieren una tarjeta controladora. El floppy disk y el/los discos rígidos pueden estar conectados en el mismo controlador o en controladores diferentes. La función del controlador es de hacer de intermediario en la transferencia de datos entre el/los discos rígidos y la memoria RAM. Se llama controlador porque contiene un chip controlador de discos rígidos, especialmente diseñado, que envía ordenes a los circuitos electrónicos para mover las cabezas, leer, escribir y efectuar otras operaciones.

A continuación mencionaremos las interfaces más usuales en controladores de discos:

• **ST506**

Los primeros discos rígidos que fueron desarrollados fueron ampliamente aceptados por el mundo de las PC's, fue el ST506. Como el nombre lo indica, los orígenes de estas controladoras proviene de la compañía de Seagate. Usualmente maneja solo dos discos rígidos. [AdEz]

• **Enhanced Small Devices Interface – ESDI**

El controlador ESDI fue desarrollado después del controlador ST506. Este controlador es un avance del controlador ST506. Generalmente, los controladores ESDI son compatibles con el ST506 y puede ser usado en computadoras cuyo BIOS está programado para soportar solamente controladores ST506. [DU99]

• **Integrated Drive Electronics – IDE**

La tecnología IDE ha progresado en estos últimos años. Los controladores IDE originalmente no soportaban más de dos discos rígidos. Esta limitación ha sido superada por la especificación de Enhanced IDE y ahora permite hasta cuatro dispositivos (discos rígidos, CD-ROMs, y Tapes) sobre una controladora. El protocolo AT Attachment (ATA) es usado para comunicarse con el controlador y los múltiples dispositivos.

Los discos IDE con mayor capacidad de 512 MB o más de 1,024 cilindros son ahora soportados sin manejadores de discos de terceras partes que rearmaban la geometría. Esto se debe al cambio que se produjo en el BIOS el cual es implementado con Lógica Block Addressing (LBA). [Ad91]

• **Small Computer System Interface – SCSI**

SCSI es una interface o mecanismo el cual permite manejar hasta ocho dispositivos (hard drive, tapes, CD-ROM, scanners) por una sola controladora. Los dispositivos SCSI deberían cumplir con al menos una de las siguientes especificaciones:

- **SCSI I.** La especificación original que provee la capacidad de tener hasta siete dispositivos sobre un bus y a un máximo de 5-Mbps de transferencia de datos, con 8 bits de datos para todos los dispositivos.

- **SCSI II.** La revisión de 1991 que incluyó el control completo del software del sistema llamado Common Command Set (CCS) e incluyó características tales como la transferencia de datos incrementado a 32 bits (llamado Wide SCSI) como una opción a incrementar la velocidad a 10 Mbps (llamado Fast SCSI). Esto habilita tener un bus para la transferencia de datos hasta 20 o 40 Mbps.
- **SCSI III.** El SCSI III es una mejora de la especificación de la SCSI II. Esta revisión separada de la revisión del CCS, adiciona al hardware tal como Serial SCSI basado sobre el P1394 (Un estándar para un serial rápido todavía en desarrollo), conexión fibra óptica y otras mejoras. SCSI III también incrementa el número de dispositivos que pueden ser conectados a un solo controlador hasta 16, y permite transmitir datos a 100 Mbps. (Dependiendo del hardware).

[Ad91]

La configuración de una SCSI bus puede ser separada en dos procesos distintos:

Configurando el controlador SCSI, tales como la asignación del IRQ, el canal DMA, etc.

Configurando el SCSI bus por sí mismo, tales como la terminación de los SCSI bus y seteando los IDs de los dispositivos.

2.4 BIOS

El BIOS (Basic Input/Output System) contiene todas las rutinas esenciales que necesita el Server para la comunicación entre el hardware y los periféricos. El BIOS puede ser encontrado en una ROM, el cual usualmente esta sobre el motherboard del Server.

• *Por qué el BIOS es importante?*

Desde que las llamadas a las rutinas han sido estandarizadas, el programador no tiene que acceder a programas para un hardware en particular. Esto significa que se puede desarrollar un programa para una Server/PC o compatible, y corre sobre otro Server/PC compatible sin ningún error, de todas maneras no significa que todas las rutinas de todos los BIOS son totalmente compatibles.

• *Arquitectura del BIOS*

El BIOS está en la ROM del Server, el cual queda residente hasta después de que el Server se apaga. La ROM que contiene el código del BIOS siempre se asigna en áreas altas de memoria, segmento F000H. La exacta asignación depende del BIOS, del sistema, y algunas veces de la capacidad de memoria. Ej: IBM BIOS comienza en E000H, mientras Phoenix BIOS puede comenzar en C000H.

El comienzo de la ROM BIOS varia según el tamaño de la ROM BIOS, pero éste usualmente termina en la última asignación de memoria de el segmento F. (dirección FFFFH).

Algunas manufacturas tienen pequeñas características agregadas al diseño de la ROM BIOS. Estas incluyen características como:

- **Shadow RAM**

Shadow RAM esta oculta en la misma dirección de memoria que la ROM BIOS. Muchos BIOS copian su código en la Shadow RAM, de donde se acceden los datos del BIOS. Esto agiliza la ejecución en la PC, porque la Shadow RAM data bus es de 16 bits, en cambio el bus de datos de la ROM BIOS es sólo de 8 bits.

- **Parámetros del Disco Rígido**

BIOS tiene frecuentemente problemas para comunicarse con los discos rígidos que existen en el mercado. Este problema es causado por los diferentes tipos de números asignados a cada disco rígido. Para que el BIOS pueda comunicarse con un disco rígido, debe conocer el número de tracks y sectores, el número de sectores por track y otros datos del disco rígido.

La solución original a este problema era una tabla de discos rígidos de la cual el usuario podía seleccionar la información del dispositivo usando el programa de Setup. Esta información debía ser pasada a la ROM BIOS. Debido a la variedad de discos rígidos que hay en el mercado esta solución quedó obsoleta. En cambio, con el programa de Setup, el usuario puede ingresar los parámetros manualmente y ésta información es pasada a la CMOS para acceder desde el BIOS.

- **Setup**

El programa Setup habilita al usuario configurar elementos de la ROM- BIOS acorde a las necesidades. Esto significa tipos de drives, date, time etc. . Algunos ROM- BIOS ofrecen la opción de configuración de la RAM como memoria expandida, EMS etc.

- **El test**

El test que realiza rutinas del BIOS consiste en el chequeo del hardware onboard del Server (procesador, memoria, interrupciones de controladoras, DMA, etc.). Si se produce algún error durante el testeo un mensaje de error o el número se despliega en la pantalla.

A continuación se da una lista de test básicos (esto es dependiente de la manufactura):

- Test de Procesador
- BIOS ROM checksum
- CMOS RAM checksum
- Test/inicialización del controlador DMA
- Test/inicialización del controlador de teclado
- Primer chequeo de 64 K de RAM
- Test/inicialización del controlador de interrupciones
- Test/inicialización del controlador de cache
- Controlador de video
- RAM arriba de los 64 K
- Interfaces Serial y Paralelo
- Controladores de discos

Una vez que el test está completado, se realiza una búsqueda de las extensiones de la ROM BIOS como por ejemplo: controladores EGA, VGA y Super VGA que tienen sus propias funciones que reemplazan las viejas interrupciones del BIOS 10H o las controladoras SCSI que no usan la interrupción 13H de disco que tiene como estándares el BIOS.

3. Accesos a dispositivos

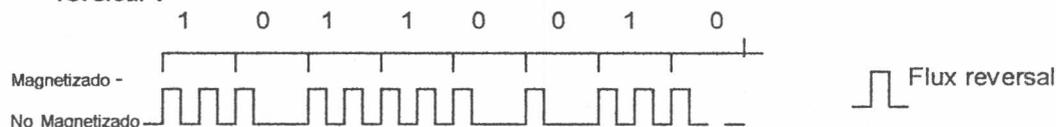
3.1 Codificación de datos

Una de las diferencias entre las controladoras es como codifican los datos. Se denomina codificación a la forma en que se graban los datos de la memoria del ordenador en la superficie del disco rígido. La definición de que un carácter es representado por una secuencia de ocho "On"/0 y "Off"/1, mayormente lleva a la interpretación de que en la pista del disco se encuentran únicamente dichas secuencias, esto no es cierto, se almacenan tantos datos en un sitio tan reducido que el controlador del disco se perdería si no se insertase información adicional entre los datos. Por esta razón, los datos tienen que ser codificados.

• Codificación FM (Modulación de Frecuencia)

Una forma simple de representar ceros y unos en un medio magnético es registrando un "flux reversal" para cada uno-bit y omitir un "flux reversal" para cada cero-bit. Sin embargo, podemos encontrar problemas cuando se registra una serie de ceros, que equivaldría a omitir "flux reversal". Este caso confundiría a las controladoras que dependen del "flux reversal" para mantener la sincronización de lo que está almacenado en el disco rígido.

Para aislar este problema se agrega una señal de clock que es escrita con los datos. Usando la técnica FM el uno-bit es registrado como dos "flux reversal" consecutivos y el cero-bit es registrado como un "flux reversal" y seguido por una omisión de un "flux reversal".



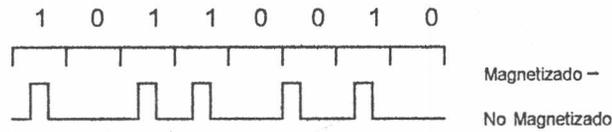
Aunque este método es simple tiene un gran desventaja. Cada bit de datos requiere dos "flux reversal", el cual reduce la capacidad del disco rígido a la mitad.

• Codificación MFM (Modulación de Frecuencia Modificada)

Para reducir la cantidad de "flux reversal" del método MF e incrementar la densidad que puede ser almacenada, otra codificación es utilizada la MFM, básicamente la codificación de los datos es codificada de la siguiente manera:

Tabla de codificación para MFM	
Valor del bit	Codificado como:
1	Flux reversal
0 seguido por otro 0-bit	Flux reversal seguido por un no flux reversal
0 seguido por otro 1-bit	No flux reversal seguido por un no flux reversal

De esta manera ambos cero-bit y uno-bit son almacenados usando un solo "flux reversal"



• **Codificación RLL (Codificación de Longitud Limitada)**

El método RLL permite almacenar mas del 50% de información que los discos que soportan la codificación MFM.

En RLL, los unos-bit son almacenados como "flux reversal" y los ceros-bit son almacenados con ausencia de "flux reversal". El método RLL en vez de codificar un solo bit, codifica un grupo de bits. Este grupo varía entre 2 y 4 bits. El esquema 2,7 RLL es el estándar de hoy, que toma como parámetros 2 y máximo 7 ceros que pueden aparecer entre dos unos.

3.2 Acceso a Discos Rígidos por medio del BIOS

El acceso a los discos rígidos por medio del BIOS se realiza a través de la interrupción 13H que comparte con el floppy disk. Aunque las funciones para el disco rígido y floppy disk son idénticas, el BIOS controla el disco rígido de una manera diferente y con módulos distintos que el de floppy drive.

Cuando la interrupción 13H es invocada, el número del dispositivo en el registro DL determina si lo que se esta direccionando es un disco rígido o floppy disk. El valor 80H representa el primer disco rígido, en cambio 81H representa el segundo disco rígido y así sucesivamente.

Las funciones de BIOS para el acceso a hard drive se muestra en la siguiente tabla:

Funciones	Tareas
00H	Reset
01H	Read status
02H	Read
03H	Write
04H	Verify
05H	Format
08H	Check format
09H	Adapt to foreign drives
0AH	Extended read
0BH	Extended write
0CH	Move read/write head
0DH	Reset
0EH	Controller read test
10H	Drive ready?
12H	Controller RAM test
13H	Drive test
14H	Controller diagnostic
15H	Determine drive type

3.3 Acceso a Discos Rígidos conectados a controladoras

Las controladoras que tiene conectados dispositivos son las que controlan y manejan el acceso a los mismos. Se puede acceder a los distintos discos invocando directamente por el BIOS y la controladora será la encargada de realizar los comandos solicitados. Puesto que los discos son dispositivos electromagnéticos, sólo son capaces de realizar órdenes algo primitivas. Las señales típicas de interfaz entre una unidad de disco y el controlador de disco de un sistema informático se encuentran en la Figura 3.3. Dado que un controlador suele ser capaz de manejar varias unidades de características similares, hacen falta unas pocas líneas de control para seleccionar una unidad designada para participar en una operación determinada. Estas se llaman líneas *Selección De Unidad* en la Figura 3.3.

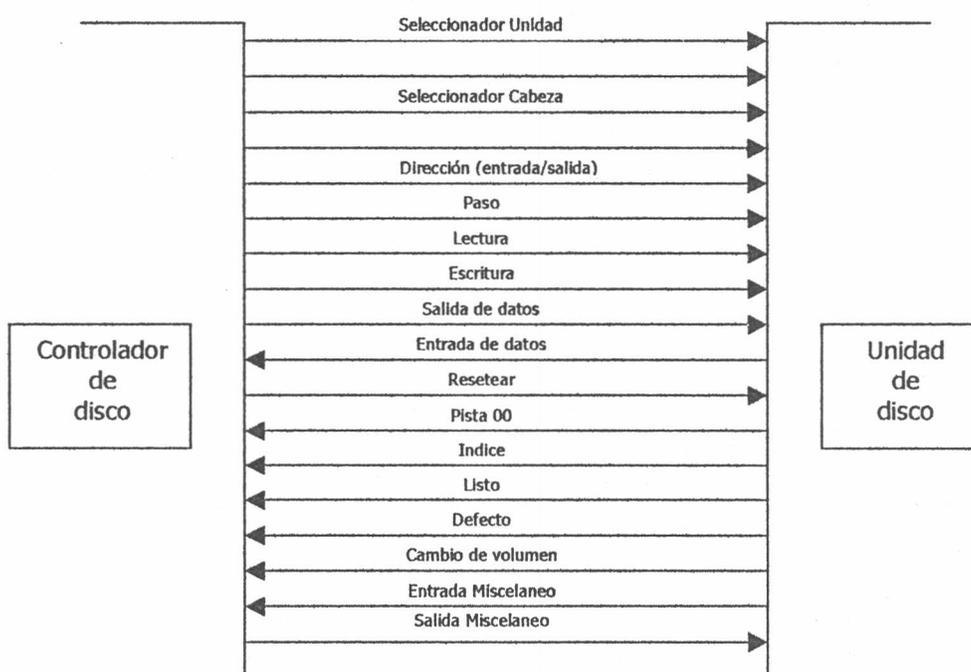


Figura 3.3

De manera parecida, las líneas *Selección De Cabeza* activan una cabeza específica en la unidad seleccionada. La señal *Dirección*, requerida para unidades de cabeza móvil, designa la dirección *Dentro* o *Fuera* del movimiento de las cabezas, partiendo de la posición actual. La línea *Step* proporciona una secuencia temporizada de pulsos de paso. Generalmente, un pulso mueve la cabeza, un cilindro y un número pre-determinado de pulsos mueven el conjunto de cabezas desde su cilindro actual hasta el cilindro destino. Las señales *Read* y *Write* activan la cabeza seleccionada de lectura o escritura. Las líneas *Entrada Datos* y *Salida Datos* suelen llevar el flujo de bits de entrada o salida cuando hay en marcha una operación *Read* o *Write* respectivamente. *Pista 00* es una señal suministrada por la unidad, que indica cuándo el conjunto de cabezas está en el cilindro 0, pista 0, la posición más lejana o cercana. La señal *Indice* indica cuándo los circuitos

de la unidad notan la marca de direccionamiento del cilindro o de la pista. La señal *Volumen Cambiado*, normalmente disponible para medios extraíbles, avisa al sistema operativo sobre cambios en los medios. Al detectar este suceso, el sistema operativo debe Invalidar en la memoria principal toda la información referente a la unidad en cuestión, tales como entradas de directorio, tablas de espacio libre, etc. Otras señales incluyen las indicaciones *Reposicionar* y *Defecto*, y otras pocas señales específicas de dispositivo, llamadas colectivamente *Entrada* y *Salida Miscelanea*. Algunos ejemplos de éstas incluyen el *Modo* de codificar *Puerta Abierta*, *Motor Encendido* y *Proteger Escritura*. [Te72]

Los dispositivos que están conectados a controladoras son los que realizan el acceso y control.

El formato en el cual los datos son salvados en un disco rígido no solo depende de la controladora sino también de la transferencia de los datos entre la computadora y el disco rígido.

A continuación mostramos la tabla de transferencia teórica de las controladoras anteriormente mencionadas:

Controladora	Máxima transferencia de datos
ST506	1 MB/seg
ESDI	2.5 MB/seg
IDE	4 MB./seg.
SCSI	5 MB./seg.

Mencionaremos la estructura, las ventajas y desventajas y el rol del BIOS en cada uno de los tipos de controladores:

• **ST506**

Generalmente los discos rígidos diseñados para ser conectados a controladoras ST506 son identificados por el rotulo "MFM/RLL". Usualmente se pueden usar DIP switches para setear el formato que va a utilizar la controladora.

Esta controladora tiene los discos rígidos conectados por medio de dos cables de 20-pin cada uno. Si los dos discos rígidos que están conectados comparten un cable de 34-pin, dicho cable se utiliza para enviar señales de control al disco rígido como por ejemplo el posicionamiento al cilindro correcto.

Los datos leídos a ser escritos en el disco rígido son transferidos sobre el cable en un manera serial y analógica.

El controlador convierte la información digital sobre un cilindro en una cadena de bit. La información sobre el medio magnético existe como valores 0 o 1. El controlador puede convertir los valores digitales como sea necesario a este proceso se llama "flux reversal".

La limitación de este tipo de controladoras es que soportan hasta dos discos rígidos, el máximo número de cilindros es de 1024, el máximo número de sectores por pista es de 63, el máximo número de cabezas es de 16, el tamaño del sector es fijo de 512 bytes, teniendo en cuenta estas limitaciones el máximo disco rígido es de 504 MB.

• **ESDI**

A diferencia de los controladores ST506, ESDI no transfiere todos los "flux reversal" serialmente, en vez, parte de la decodificación lógica se encuentra en el disco rígido llamado *separador de datos*. El separador de datos prepara el dato leído del disco rígido y lo transfiere, solamente el dato útil, en una forma digital al controlador.

Porque el controlador y el separador de datos trabajan paralelamente, la transferencia puede ser alcanzada hasta 10 meg/seg. No solo el separador de datos es un componente inteligente del tipo de controlador ESDI, sino que también almacena información acerca del formato físico y de sectores defectuosos.

La información del disco rígido es almacenada en la CMOS de una AT. Como las controladoras ESDI pueden requerir la información al disco rígido, puede ser que la información que tiene el BIOS no coincida con las características del hard drive (por ejemplo limitación de la lista de discos rígidos que tenga el BIOS). El problema de no coincidir las características del BIOS con el disco rígido es que se puede perder parte de la capacidad del disco rígido. Afortunadamente la mayoría de las controladoras ESDI pueden aislar este problema. Si las características registradas en el BIOS es menor a las características del ESDI disco rígido, las controladoras ESDI utilizan características especiales llamadas *translación de sector*, el controlador ESDI convierte las especificaciones lógicas del BIOS en las especificaciones físicas del disco rígido. La característica de translación de sector sólo depende de las controladoras ESDI.

• **IDE**

Usando la interface IDE el disco rígido está conectado por un cable de 40-pin que combina la función de cable para datos como para control y está directamente conectado al motherboard. Los BIOS actuales soportan disco rígido con interface IDE.

• **SCSI**

Los comandos para el control de dispositivos SCSI están estandarizados, el cable que une a los dispositivos SCSI es de 80-pin.

Los sistemas SCSI tienen su propio BIOS. Las funciones originales de BIOS para el acceso a los discos son remplazados por el BIOS del SCSI. El manejo de los dispositivos se realiza por el driver correspondiente a la controladora.

Como mencionamos anteriormente la principal función de un controlador de disco es la de convertir las órdenes de nivel más alto, tales como las de *Seek* o *Read* un sector, en una secuencia correctamente temporizada de órdenes específicas de unidad. Además, el controlador proporciona la conversión en serie/paralela y el acondicionamiento de señales necesario para convertir el formato de byte o palabra requerido para la comunicación DMA con la memoria principal, en los flujos análogos de series bits, esperados y producidos por las unidades de disco.

Otra función importante de los controladores de disco es el control de errores.

El control de errores es necesario para la detección de errores transitorios, causados por ruidos y la interferencia electromagnética, y para el manejo de defectos en los medios. Siempre que se vaya a escribir un bloque de datos en disco, el controlador calcula los valores de los datos y adjunta los bits de comparación a los mismos. Puesto que los errores en disco ocurren a menudo en ráfagas que producen errores encadenados de bits sucesivos, se usan a menudo en los controladores de disco

mecanismos para la detección de errores, capaces de encargarse de este tipo de error, tales como códigos cíclicos de redundancia (CRC). Durante las operaciones de Read, el controlador calcula los bits de comprobación, basándose en los datos recibidos. Se verifica la exactitud de los datos recibidos comparando los bits de comprobación calculados con los recibidos del disco. Cualquier discrepancia indica un error. Según su causa, un error en disco puede ser transitorio o permanente. Para contrarrestar los errores transitorios, se suele re-leer el sector en cuestión un número predeterminado de veces. Este parámetro se llama a menudo el *contador de reintentos*. Si el error persiste después de unos pocos re-intentos, se lo asume permanente. Se suele llamar a los sectores defectuosos *bloques malos*. Una capa de bajo nivel del sistema del control de archivos suele encargarse de evitar los bloques malos, para que no tenga que implicarse el usuario. Los bloques malos pueden ser detectados a la hora de inicializar el disco.

3.4 Optimización de la búsqueda en un disco rígido

La búsqueda de información en un disco rígido tiene que realizarse de la manera más óptima, de esta manera tanto el backup como en el restore se realizará en un menor tiempo para obtener el sistema en producción lo antes posible. Para ello estudiamos los distintos modos de búsqueda o acceso al disco.

El acceso a un disco (lectura o escritura) consta de tres acciones significativas, a saber, una búsqueda, un retraso rotacional (latencia) y una transmisión del registro. Algunas de las estrategias más comunes son:

Planificación FCFS (primero en llegar, primero en ser servido)

En la planificación FCFS, la primera petición que llega es la primera en ser servida. FCFS es justa en el sentido de que una vez que llega una petición, se fija su lugar dentro de la planificación. Una petición no puede ser desplazada por la llegada de una petición con prioridad más alta.

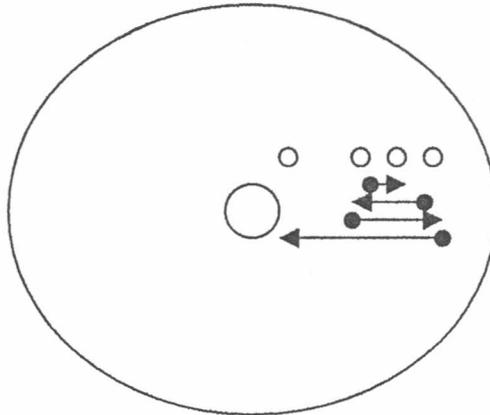
Cuando las peticiones se encuentran uniformemente distribuidas sobre las superficies de los discos, la planificación FCFS da como resultado un patrón de búsqueda al azar que ignora las relaciones posicionales entre las peticiones pendientes de la cola y no hace ningún intento de optimizar el patrón de búsqueda.

FCFS es aceptable cuando la carga en un disco es ligera. Pero a medida que crece la carga, FCFS tiende a saturar el dispositivo y los tiempos de respuesta se incrementan. FCFS ofrece una varianza pequeña, pero esto sirve de poco consuelo a la petición situada en la última parte de la cola del disco, mientras que la barra vaga de un lado a otro, en una acalorada "danza del disco". [Ha87]

Planificación SSTF (menor tiempo de búsqueda primero)

En la planificación SSTF, la petición que da por resultado la distancia de búsqueda más corta (y, con esto, el tiempo de búsqueda más corto) es la siguiente en ser servida, aunque esa petición no sea primera en la cola. SSTF tiende a discriminar de manera clara ciertas peticiones. Los patrones de búsqueda de SSTF tienden a estar muy localizados, dando como resultado que las pistas más internas y externas reciban un pobre servicio, en comparación con las pistas del centro.

SSTF da como resultado mejores tasas de capacidad de ejecución que FCFS, y la media de tiempos de respuesta tiende a ser más baja, para cargas moderadas. Una desventaja significativa es que tiene lugar varianzas más altas debido a la discriminación de las pistas interiores y exteriores. Cuando se consideran las mejoras significativas en la capacidad de ejecución y en la media de los tiempos de respuesta, éste



aumento de la varianza de los tiempos de respuesta (es decir, su falta de predecibilidad) lo hacen inaceptable para los sistemas interactivos. [Ha87]

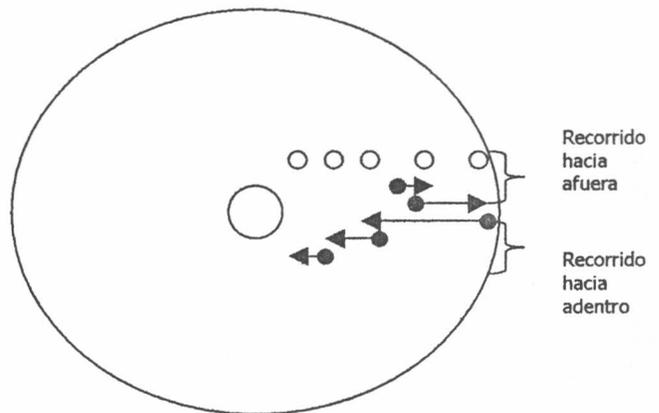
Planificación SCAN

Denning desarrolló la estrategia de planificación SCAN para solucionar los problemas de discriminación y alta varianza de los tiempos de respuesta de SSTF.

SCAN opera como SSTF, excepto que selecciona la petición que da como resultado la distancia de búsqueda más corta en una dirección preferida, si la dirección preferida es en ese instante hacia afuera.

SCAN no cambia la dirección hasta que ha alcanzado el cilindro exterior o hasta que ya no haya peticiones pendientes en la dirección preferida. SCAN ha sido la base de la mayoría de las estrategias de planificación que se implementan en la actualidad.

SCAN se comporta de manera muy parecida a SSTF desde el punto de vista de la mejora de la capacidad de ejecución y de la media de los tiempos de respuesta, pero elimina mucha de la discriminación inherente a los esquemas SSTF y ofrece una varianza mejor.

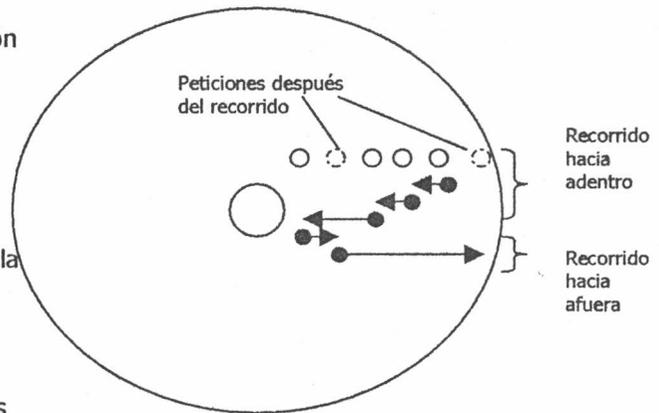


Debido al movimiento de oscilación de las cabezas de lectura-escritura en SCAN, las pistas exteriores son visitadas con menos frecuencia que las pistas intermedias, pero esto no es tan grave como la discriminación del SSTF. [Ha87]

Planificación SCAN de n-pasos

Una interesante modificación de la estrategia SCAN básica se denomina SCAN de n-pasos. En esta estrategia, el brazo del disco se mueve de un lado a otro como en SCAN, pero sólo da servicio a aquellas peticiones que se encuentran en espera cuando comienza un recorrido particular. Las peticiones que llegan durante un recorrido son agrupadas y ordenadas para un servicio óptimo durante el recorrido de regreso.

SCAN de n-pasos ofrece un buen rendimiento de la capacidad de ejecución y de la media de los tiempos de respuesta. Su característica más significativa es una menor varianza de los tiempos de respuesta que en las planificaciones SSFT y SCAN convencionales. SCAN de n-pasos evita la posibilidad de postergación indefinida que tiene lugar si un gran número de peticiones llegan al cilindro que está siendo servido y guarda estas peticiones para ser servidas durante el recorrido de regreso. [Ha87]

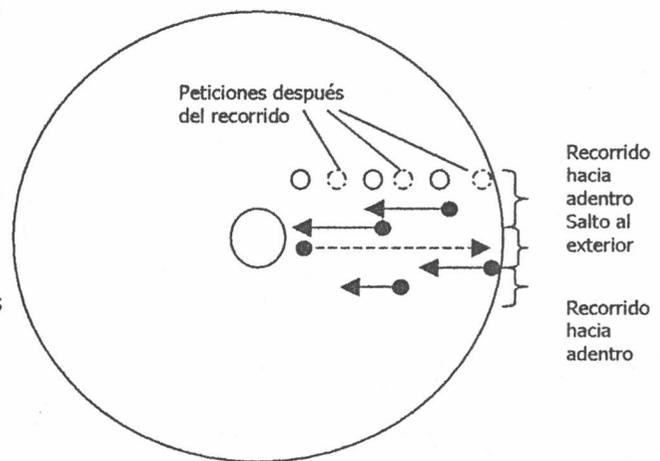


Planificación C-SCAN

Otra interesante modificación a la estrategia SCAN básica se denomina C-SCAN (por SCAN circular). C-SCAN elimina la discriminación de estrategias anteriores contra los cilindros exteriores e interiores.

En la estrategia C-SCAN, el brazo del cilindro se mueve del cilindro exterior al interior, sirviendo a las peticiones sobre una base de búsqueda más corta. Cuando el brazo a completado su recorrido hacia adentro, salta a la petición más cercana al cilindro exterior y a continuación reanuda su recorrido hacia adentro procesando peticiones.

C-SCAN puede implementarse de manera que las peticiones que llegan durante un recorrido sean servidas en el siguiente. De esta forma C-SCAN elimina completamente la discriminación contra las peticiones para los cilindros exterior e interior. Tiene una varianza de los tiempos de respuesta muy pequeña. Los resultados de simulación presentados en la literatura [Te 72a] indican que la mejor política de planificación de disco puede operar en dos etapas. Con carga baja, la política SCAN es la mejor,



mientras que con cargas medias y pesadas, C-SCAN produce los mejores resultados. C-SCAN con optimización rotacional maneja de manera efectiva condiciones de cargas muy pesadas. [Ha87]

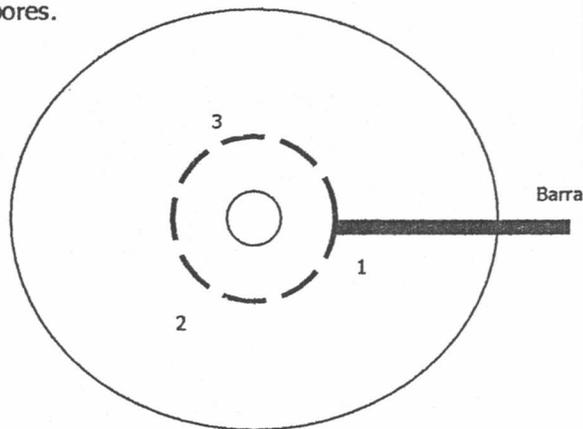
Esquema Eschenbach

El movimiento del brazo del disco es circular como en C-SCAN, pero con varias excepciones importantes. Cada cilindro es servido exactamente por un pista completa de información, haya o no peticiones para ese cilindro. Las peticiones se reordenan para ser servidas dentro de un cilindro para tomar ventaja de la posición rotacional, pero si dos peticiones trasladan posiciones de sectores dentro de un cilindro, solamente se sirve una en el movimiento actual del brazo del disco. [Ha87]

Optimización Rotacional

En condiciones de carga pesada, las probabilidades de que ocurran referencias al mismo cilindro aumentan y resulta útil considerar la optimización rotacional y la optimización de búsqueda. La optimización rotacional se ha utilizado durante muchos años en dispositivos de cabeza fija como tambores.

Una forma paralela a la estrategia de optimización de la búsqueda SSTF es la estrategia SLTF (tiempo de latencia más corto primero) de optimización rotacional. Una vez que el brazo llega a un cilindro en particular, pueden existir muchas peticiones pendientes en las distintas pistas de ese cilindro. La estrategia SLTF examina todas estas peticiones y sirve primero a aquella que tiene el retraso rotacional más corto.



Esta estrategia ha demostrado estar muy cerca del óptimo teórico y es relativamente fácil de implementar. [Ha87]

4. Sistemas Operativos

El sistema de archivos debe aceptar las peticiones de usuarios de esta forma, y traducirlas en direcciones físicas de tres componentes comprensibles para la controladora del disco, los números del sector, cabeza y cilindro de una unidad determinada.

Aunque la correlación directa lógico-físico de las direcciones en disco es posible, la mayoría de los sistemas la realizan en etapas. Una de las razones principales por hacerlo así es la gran variabilidad de estructura y de otras características físicas de distintos dispositivos de almacenamiento secundario. Incluso los dispositivos similares, como las unidades de disco, pueden variar en el número de cabezas de lectura/escritura, etc. El soporte de tal variedad mediante la correlación lógico-física de direcciones requería la incorporación de conocimientos de cada dispositivo de almacenamiento específico en casi cada capa del

sistema de la gestión de archivo. Puesto que el sistema de la gestión de archivos suele ser uno de los módulos más amplios de un sistema operativo, esto resultaría en un diseño engorroso y no estructurado. Quizás un problema aún más serio sea la falta de flexibilidad, que dificultaría la incorporación de nuevos tipos de unidades en semejante sistema, después de la terminación de su capa de gestión de archivo.

Para evitar estos problemas, los diseñadores del sistema operativo suelen seguir un método más estructurado. El sistema de archivos suele implementarse en la forma de varias capas, cada una de las cuales aportando su propia abstracción de dispositivos de almacenamiento secundario. Sólo las capas más bajas de esta estructura, como las controladoras de dispositivos, están explícitamente enteradas de las características del dispositivo físico. Las capas más altas operan con abstractos dispositivos virtuales, que engloban las características comunes de los verdaderos dispositivos de almacenamiento, tales como la direccionabilidad orientada a bloques y la transferencia de datos. Las capas de bajo nivel, los drivers de dispositivos, por ejemplo, traducen este enfoque al verdadero conjunto de operaciones específicas a dispositivo. Se logra de esta manera un diseño mucho más manejable y estructurado. Además, es bastante fácil añadir nuevos tipos de dispositivos al sistema, escribiendo sólo una conducta de dispositivo relativamente corta y dejando intactas las capas superiores del sistema de archivos.

Aunque varíen los detalles, hay tres niveles de abstracción de dispositivo comúnmente identificables en las implementaciones del sistema de la gestión de archivos. El primero es cómo el usuario ve los archivos denominados, consistentes en colección de bytes o registros. Los usuarios dirigen la información en archivos en términos de direcciones lógicas relacionadas con archivos.

La segunda, más baja, abstracción considera cada volumen del almacenamiento un conjunto lineal de bloques lógicos. Las direcciones a este nivel suelen llamarse direcciones lógicas relativas a volumen. El tamaño del bloque lógico suele fijarse a nivel de sistema. El sistema suele inicializar volúmenes con tamaños de sector elegibles por el software, para que los sectores tengan el mismo tamaño que el bloque físico. Se manejan los dispositivos con tamaños de sector o de bloque fijos pero diferentes, al hacer la correlación de bloque lógico-físico a nivel drivers. Esto suele ser una operación sencilla, puesto que la mayoría de los tamaños de bloque y sector físicos son potencias enteras de base 2.

La tercera abstracción de los dispositivos de almacenamiento es la proporcionada por los drivers de dispositivo. A este nivel, los discos son direccionados mediante las direcciones físicas de tres componentes. Como ya hemos descripto, los drivers de dispositivo las traduce a las señales reales de control de unidad, como las representadas en la figura 3.3.

Además de direccionar las traducciones entre distintos niveles de la abstracción de disco, el sistema de la gestión de archivos debe encargarse de la posible discrepancia que resulta de la organización en bloques fijos de algunos dispositivos de almacenamiento y de la supresión de esta consideración desde el punto de vista del usuario. La petición de un usuario puede traducirse en un número fraccional de bloques, mientras que la mayoría de los dispositivos de almacenamiento sólo son capaces de transferir números enteros de bloques. Además, los usuarios pueden direccionar bytes individuales en un archivo, mientras que los discos sólo pueden direccionar unidades a nivel bloque. ¿Qué ocurre cuando un usuario desea tener un acceso a un sólo byte localizado en medio de un bloque de disco? Los sistemas de archivos suelen solucionar este problema manteniendo almacenamientos intermedios internos, desde los que se intercambian datos en bloques con los dispositivos de almacenamiento

secundario. Luego se extraen de estos bloques por medio de operaciones lógicas las unidades específicas pedidas por los usuarios, y se las transfiere al espacio del usuario. Este proceso se llama bloqueo y desbloqueo, según si los datos están empaquetados en bloques para salida o desempaquetados después de su entrada desde un dispositivo de almacenamiento secundario.

TRABAJO REALIZADO:
"BCR UN SISTEMA DE BACKUP Y RESTORE EN CASO DE CATASTROFE"

El "BCR" es un software que esta diseñado para realizar backups y restore en caso de catástrofe (destrucción total o parte de la información).

El resguardo que realiza el "BCR" es una imagen del disco a nivel físico en un almacenamiento externo. En caso de catástrofe se utilizará este resguardo para restaurar el sistema al estado en el que se encontraba en el momento de realizarse el backup. Una vez obtenido dicho estado consistente el operador podrá utilizar las herramientas diarias de backup que están diseñados para resguardar la información a nivel de archivos y sistema de directorio, y de esta forma lograr un estado con la mínima perdida de información con una rápida puesta en operación del sistema sin necesidad de reinstalación de ningún tipo de software (S.O., software de base, software de backup, drivers, etc).

Para realizar la imagen de los discos es necesario conocer la geometría de los mismos y realizar el backup OffLine para asegurar la consistencia de la información.

El "BCR" inicia el sistema con un sistema operativo independiente del que se utiliza en el equipo, esto nos permite utilizar el "BCR" sobre cualquier plataforma de sistema operativo y obtener un backup OffLine.

Las características en la que esta basado el "BCR" que fueron descriptas anteriormente son las siguientes:

- Realización del Backup Offline
- Conocimiento de la geometría de los discos
- El acceso al disco rígido esta basado en la estrategia de Optimización Rotacional

Durante la etapa de investigación determinamos que se requería la configuración geométrica de los disco, que puede ser realizada manualmente o con alguna detección automática. Lo que llegamos a observar es que ha existido una evolución en la definición de los parámetros de los discos que determina su geometría a un formato estándar, almacenado y manejado por el BIOS. Una vez establecido la geometría del/los disco/s teníamos que acceder al disco para leer y escribir la información, para ello analizamos las funciones del BIOS y verificamos que nos permite acceder a distintos tipos de discos tomando en cuenta el tamaño y la interface (IDE, SCSI).

El siguiente punto que abordamos fue el elegir la estrategia de acceso al disco, tomando como premisa que el objetivo es realizar un backup completo del disco, y probando varias alternativas, que describimos en el punto 3.4 ,decidimos implementar la estrategia de optimización rotacional.

Para definir la funcionalidad del BCR analizamos las distintas alternativas o combinaciones en cuanto a hardware y software (cantidad de discos, medios de almacenamientos, tipos de discos, controladoras).

También analizamos casos especiales como ser configuraciones de tipo RAID (0,1,5), en el caso de RAID 0,5 se deberá realizar el backup de todos los discos que componen el RAID, pues la información se encuentra distribuida de una manera dependiente, no así en RAID 1, en donde será suficiente hacerlo en uno solo de los discos, pues en ambos se encuentra idéntica información.

Para determinar la consistencia de la información lo que consideramos fue la presencia de errores físicos en los discos rígidos:

Un caso es que al estar realizando el backup, se detecte un sector dañado, si el BCR guardara con la confirmación del operador, este sector como información neutra, el respaldo no sería consistente siempre y cuando originalmente dicho sector almacenara información. Entonces la recomendación es utilizar herramientas provistas por el sistema operativo utilizado o terceros que aseguren que los sectores dañados, si es que hubieren, estén debidamente manejados por el S.O. para que de esta forma no contengan información.

Otro caso es cuando el proceso de backup se realizó sin inconvenientes, pero al realizar la restauración el BCR detecta un sector dañado en el disco destino, en este caso se recomienda cambiar dicho disco, pero se dejará la alternativa al operador para continuar dicha operación pues no se puede establecer en este momento si dicho sector contiene información o no.

Luego diseñamos las bases de datos necesarias para almacenar las características de los discos rígidos y los medios de almacenamiento externos, como así también para controlar la secuencia y consistencia de los medios de respaldo.

La evolución de los medio magnéticos de almacenamiento externo, nos lleva a diseñar una estructura que nos permita incorporar los correspondientes drivers sin modificar la programación del BCR.

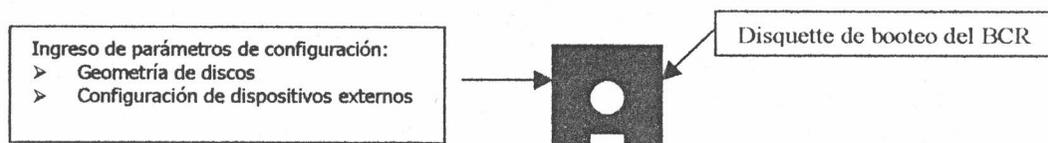
1. Esquema General del BCR

Podríamos dividir el proceso de protección de la información en caso de catástrofe con BCR en tres etapas:

- Configuración de dispositivos
- Realización de backup
- Realización de restore

Configuración de dispositivos

En esta etapa se ingresaran en "BCR" los parámetros necesarios como ser la geometría de los discos, la definición e instalación de driver del dispositivo externo donde se realizara el backup/restore y se almacenara en el disquette de booteo del "BCR", que se utilizará para realizar los procesos de backup/restore.



Realización de backup

En esta etapa se debe bootear el equipo con el disquete generado del "BCR" y una vez presentado el menú del sistema seleccionar el/los disco/s a resguardar y además el medio externo en el cual se realizara el backup, después de confirmar lo seleccionado el "BCR" comienza a realizar el resguardo.

Para explicar un proceso completo de backup de "BCR" tomaremos como ejemplo la realización del backup de un disco con la siguiente geometría:

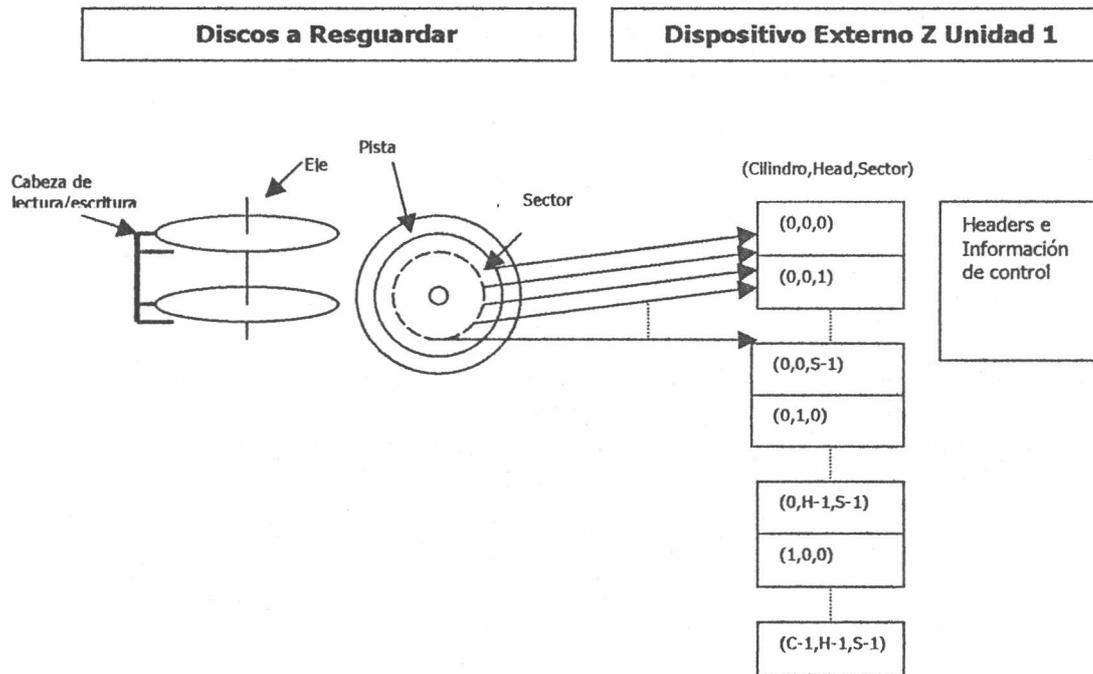
Cilindro= C

Head=H

Sector=S

Dispositivo externo: Z

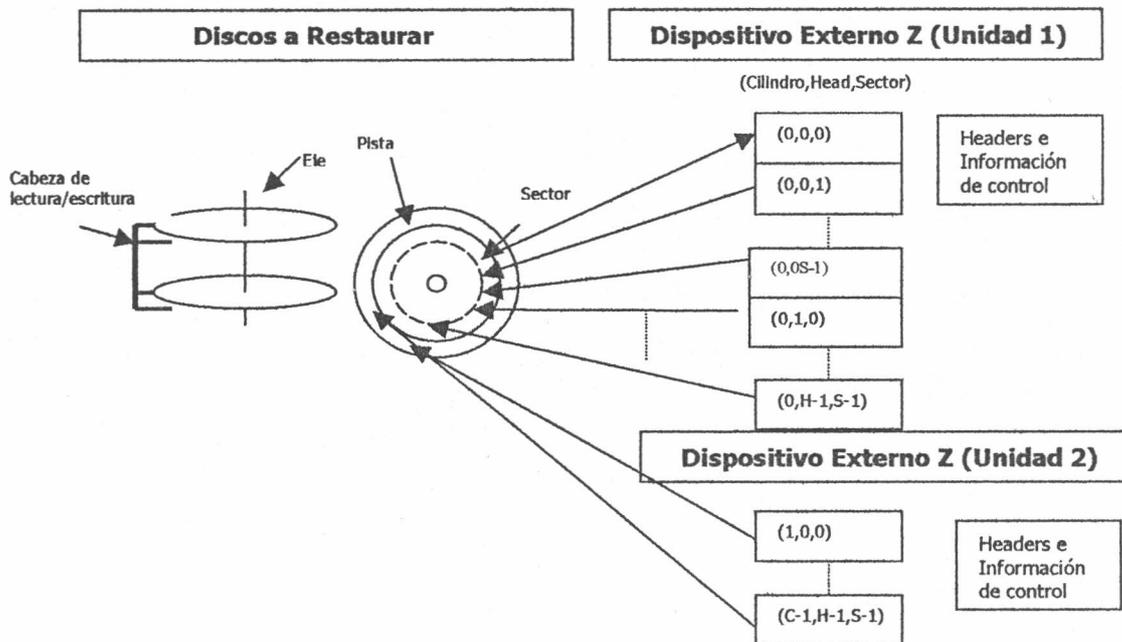
Denominaremos "Dispositivo externo: Z" donde se almacenará el backup, por razones de costos y dispositivos disponibles en nuestras pruebas fueron seleccionados como dispositivos externos, un Zip drive IDE Iomega, un Jaz drive SCSI Iomega y un disco rígido IDE, pero pudiendo seleccionar en un futuro CD, Cartridge, etc.



El proceso de resguardo de BCR recorre el disco utilizando la estrategia de optimización rotacional resguardando la información en el dispositivo externo en forma secuencial, además almacena el header con la información de secuencia y control.

Realización de restore

En esta etapa se debe bootear el equipo con el disquete generado del "BCR" y una vez presentado el menú del sistema seleccionar el/los disco/s a restaurar y además el medio externo en el cual se encuentra el backup correspondiente, después de confirmar lo seleccionado el "BCR" realiza un control de que lo seleccionado corresponda a la información almacenada en el dispositivo externo utilizando el header del medio externo y las tablas del disco de configuración "BCR" y comienza a realizar la restauración.



El proceso de restauración de BCR recorre el dispositivo externo en forma secuencial y utilizando la información del header y de las tablas ingresadas en el disco de configuración "BCR", va almacenando en el disco la información. Además utiliza la información de secuencia y control del header para el caso en que un disco este resguardado en mas de una unidad del dispositivo externo.

2. Consideraciones para obtener una restauración consistente

- Realizar backup con el "BCR" cuando el sistema se encuentre en un estado "consistente".
- Realizar backup con el "BCR" cuando se modifica la configuración del sistema, como por ejemplo la modificación del hardware, cambio de configuración del S.O., instalación de nuevas aplicaciones, etc...
- Regenerar el disco de "BCR" cuando se modifiquen las configuraciones del subsistema de discos y dispositivos de almacenamiento externos.
- Si su configuración de hardware no tiene implementado ningún nivel de RAID.
 - Si su sistema esta compuesto por un solo disco: Realizar la restauración del mismo.
 - Si su sistema esta compuesto por 2 o mas discos tiene las siguientes opciones:
 - a.) Realizar la restauración de los todos los discos del sistema.
 - b.) Dependiendo de la información almacenada en los discos y identificando localizando el/los disco/s con problemas, se puede realizar una restauración parcial.
- Si su configuración de hardware tiene implementado algún nivel de RAID
 - Si su sistema tiene implementado RAID 0 (Stripe Sets) : Deberá restaurar todos los discos del sistema.
 - Si su sistema tiene implementado RAID 1 (Mirror Sets) : Realizar el Backup del disco Master, en caso que el problema sea lógico, restaurar el disco master. En caso de problemas con el hardware de uno de los discos realizar la recuperación con la metodología de Mirror Set, en el caso de problemas con los dos discos, realizar la restauración del disco master.
 - Si su sistema tiene implementado RAID 5 (Stripe Sets With Parity) : Deberá restaurar todos los discos del sistema.
- Después de utilizar la restauración con el "BCR" se recomienda utilizar la herramienta de backup diaria para restaurar la información mas reciente.

3. Ejemplo de un escenario en caso de catástrofe

Escenario : La empresa ACME dispone de una red con un servidor COMPAQ con sistema operativo WINDOWS NT con un disco rígido SCSI de 3 GB. El mismo de utiliza para proveer a los usuarios de aplicaciones comerciales. Además dispone de una unidad externa de backup de 2GB. La empresa no posee de un sector de informática propio.

SUCESO: Un virus Informático borra la FAT del servidor.

ACCION SIN BCR (ESCENARIO PESIMO): Se debe contratar un servicio externo para la solución de la catástrofe, el cual se encuentra con los siguientes problemas:

- No se encuentran los CD's de instalación de WINDOWS NT.
- No se tiene registrado la configuración de Windows NT.
- El Backup solo contiene los datos del sistema de aplicación.
- Los CD's que contienen el software de aplicación no coinciden con los instalados en el sistema.
- La empresa proveedora del software de aplicación no se encuentra más en el mercado.

- No se encuentran los drivers de las placa SCSI, placa de red, medio externo, software de backup, etc..

Ante todo esto la empresa contratada aconseja de comprar una nueva versión de WINDOWS NT, reinstalar y reconfigurar todo el sistema y además comprar un nuevo sistema de gestión, lo cual demorara en adaptarse al sistema de trabajo en alrededor de 60 días.

ACCION SIN BCR (ESCENARIO OPTIMO): Se debe contratar un servicio externo para la solución de la catástrofe el cual debe realizar las siguientes tareas:

- Reinstalar el sistema operativo.
- Configurar el sistema operativo. (drivers, permisos, dispositivos, etc..)
- Reinstalar software de BASE.
- Configurar software de BASE.
- Reinstalar el software de Backup.
- Realizar la restauración con el último backup realizado.
- Comprobar el correcto funcionamiento del sistema

Tiempo estimado para solucionar el problema 24/48 Horas.

ACCION CON BCR (EN CUALQUIER ESCENARIO) : No debe contratar un servicio externo para la solución de la catástrofe pues cualquier operador puede realizar las siguientes tareas:

- Bootear con el disco de "BCR"
- Seleccionar el disco a Recuperar e ingresar la unidad en el medio externo.
- Retirar disco "BCR" y re-bootear el equipo.
- Realizar la restauración con el último backup realizado.

Tiempo estimado para solucionar el problema 1-2 Horas. Y sin ningún costo adicional.

4. Descripción del BCR

A continuación describiremos los aspectos más relevantes del sistema BCR, sin embargo se anexa el análisis funcional y la descripción de procesos para mayor detalle, como también el código fuente.

El BCR esta desarrollado en Borland C para obtener una mejor portabilidad del código y funciones que nos permite trabajar en bajo nivel accediendo a dispositivos de hardware. El sistema BCR esta dividido básicamente en funciones de validación, ingreso de configuraciones, resguardo, restauración e informes. El BCR tiene definido estructuras que son necesarias para guardar las configuraciones de discos, dispositivos externos, como también para controlar los procesos de backup y restore.

El BCR esta diseñado para que pueda respaldar cualquier sistema operativo y las aplicaciones instaladas en ellos, es por eso que el BCR se inicia desde un diskette booteable, el cual contendrá además del ejecutable del sistema (BCR.EXE), los archivos con la información de

la geometría de los discos a ser respaldados (DISCOS.DAT) , información de dispositivos externos (DISP_EXT.DAT) y los drivers correspondientes.

La primera etapa de configuración es para definir la geometría de los discos y el dispositivo externo. En esta etapa se utilizan las funciones altaDisco(), bajaDisco() para el ingreso de las geometrías de los discos o la baja de los mismo y para la asignación de dispositivo externo contamos con la función dispExt().

Las funciones altaDisco() y bajaDisco() realizan lecturas y escrituras sobre el archivo de datos Discos.dat que almacena la información de los discos que serán respaldados, el archivo respeta una estructura definida donde se almacenan las configuraciones de los discos con los campos necesarios para realizar los procesos de backup y restore. La estructura consta de campos como ser numero de disco, descripción, cantidades de head, cilindros y sectores, esta información es para cada disco ingresado, hacemos notar que también se tiene un campo que indica si ese disco fue dado de baja de la lista de discos posibles a ser respaldados. La baja de un disco es lógica, de esta manera se puede contar con información de posibles cambios de hardware que se produjeron sobre el servidor.

La función dispExt() es utilizada para la definición del dispositivo externo donde se almacenará o se restaurará la información, esta información es almacenada en el archivo Disp_Ext.dat

La segunda etapa es la de definir los discos a ser respaldados y el resguardo de los mismos. Contamos con la función backup() que primeramente valida que los discos a respaldar correspondan a los discos físicos y luego realiza el proceso de respaldo, en este proceso se crea el archivo de control (control.dat) donde se guarda el número de disco que se esta respaldando, el número de secuencia del actual medio y del siguiente, en caso que existiera. A continuación se comienza a leer el disco utilizando funciones de BIOS, el método seleccionado que aplicamos para la lectura del disco es la de Optimización Rotacional puesto que debemos leer toda la información de cada cilindro. Esta información se va almacenando en forma secuencial en el archivo backup.dat, el cual puede existir en distintos medios (cinta, zip disk, etc) dependiendo de la capacidad de los mismos. En esta etapa hay varios factores a tener en cuenta como por ejemplo la existencia de sectores dañados en los discos, el control de la secuencia de los medios de backup, la posibilidad de respaldar mas de un disco y además que tengan características diferentes.

El proceso de restauración es realizado por la función restore(), la cual controla que los discos seleccionados para la recuperación de información existan físicamente y además que se encuentren los archivos de backup (backup.dat) que les corresponden. Dichos archivos se leen en forma secuencial, y utilizando nuevamente las funciones del BIOS, pero esta vez para escribir, se van grabando sobre los discos seleccionados. La información de control disponible en los medios de respaldo (control.dat) se utiliza para el control de la secuencia y consistencia del respaldo.

La interfase con el usuario fueron diseñadas de tal manera que el sistema pueda ser utilizado por usuarios con poca experiencia.

5. CONCLUSIONES

A continuación analizaremos los objetivos planteados para el desarrollo de nuestro trabajo y ver de que manera el "BCR" los cumple:

El primero objetivo:

- ❖ Diseñar un mecanismo de backup y restore que sea independiente del sistema operativo en uso, la plataforma utilizada y la información almacenada.

EL "BCR" cumple con este objetivo dado que inicia el equipo con un sistema operativo independiente y realiza el backup a través del BIOS con el conociendo de la geometría de los discos, obteniendo un acceso directo al mismo, con estas características el "BCR" se independiza del sistema operativo que se encuentra instalado en el equipo y el formato de la información almacenada.

El segundo objetivo:

- ❖ Diseñar un mecanismo de backup y restore que permita en caso de "catástrofe" volver a un estado operable, seguro y consistente.

Para obtener una restauración consistente hay que tomar un backup de la información cuando esta se encuentra en estado consistente, para ello el "BCR" realiza el backup en modo Offline lo que permite que el único que tiene acceso a la información almacenada es el "BCR". Para obtener la restauración se debe mantener las mismas características de hardware para que no sufra ningún cambio el sistema operativo y las aplicaciones instaladas.

El tercer objetivo:

- ❖ Diseñar un mecanismo de backup y restore que requiera bajos conocimientos por parte del operador.

Para realizar el backup y el restore se utilizó un interfase guiada por elección de opciones para que el operador no tenga conocimientos avanzados para realizar el backup y la restauración del server en caso de catástrofe.

El BCR requiere una configuración inicial que puede necesitar un operador con conocimiento técnicos mas avanzado pero esto sería por una única vez de todas maneras teniendo la información adecuada también podría ser configurado por un operador normal.

6. Trabajos a Futuro (Proyecciones)

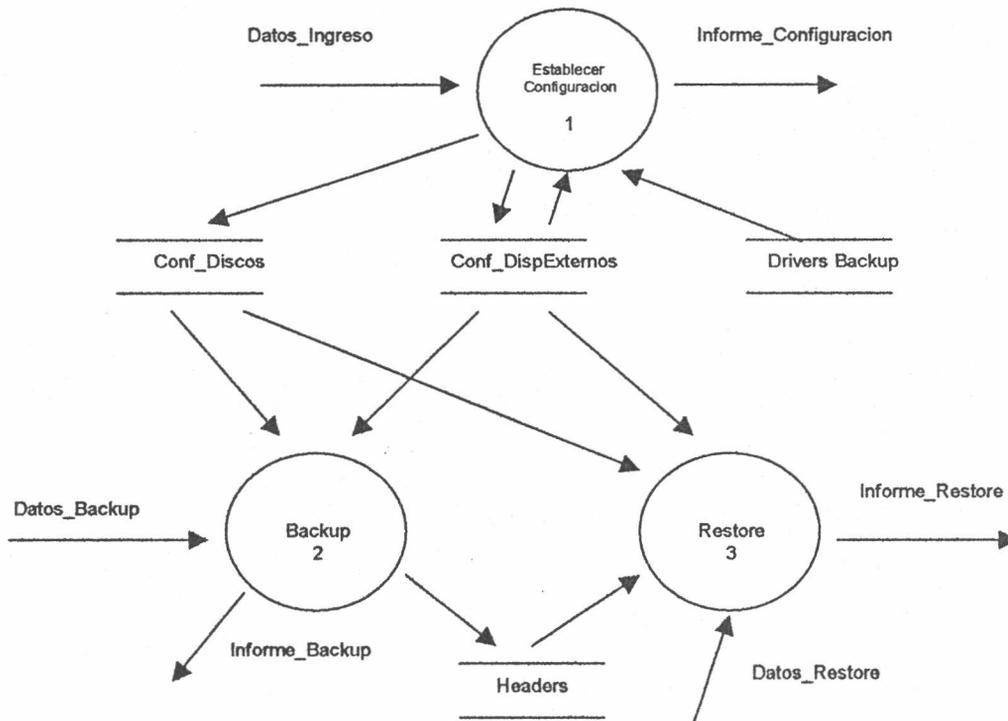
El entorno de nuestro trabajo lo limitamos a la plataforma Intel y las pruebas por razones de costos lo acotamos a determinadas tecnologías como mencionamos anteriormente en este documento; sin embargo, nuestro trabajo queda abierto para los siguientes trabajos futuros:

- **Compresión de datos.**
A través de una rutina de compresión aplicada a la información a resguardar, se podría minimizar el tiempo del proceso de backup y el espacio para almacenar la información a respaldar, esta rutina debería estar acompañada por una descompresión aplicada al proceso de restauración que agregaría un retraso de la recuperación de los datos.
- **Verificación de consistencia de Backup.**
Sería opción optativa del proceso de backup, la cual compararía el resguardo con la información real. Esta opción incrementaría el tiempo del proceso del backup, pero daría una mayor confiabilidad al resguardo realizado.
- **Estandarización de drivers para medios externos de Back-Up.**
Se podría definir un estándar o requerimientos que tendría que cumplir los drivers de los dispositivos externos donde se realizaría el Backup, para que empresas proveedoras de dispositivos de almacenamiento externos realicen los drivers de sus productos para poder ser utilizados con el "BCR".
- **Para optimizar la configuración inicial del BCR se podría evaluar la posibilidad de realizar una autodetección del hardware.**
- **Generación de Código para distintas arquitecturas y marcas de procesador.**
El código del BCR esta realizado en lenguaje C para que pueda ser transportado mas fácilmente a otras arquitecturas y utilizando los comandos mas estándares de acceso a dispositivos, lo que se podría agregar en un futuro es un generador de BCR según la arquitectura en la que se va usar.
- **Lograr que se encuentre incorporado en el BIOS como una opción dentro del menú en vez de un producto independiente.**
Se podría trabajar sobre la implementación del BCR directamente como una función del BIOS, no solamente para servidores sino también para computadoras personales donde podría tener un CD grabador donde el usuario pueda realizar su backup completo, pudiendo respaldar y restaurar rápidamente su información en caso de catástrofe.

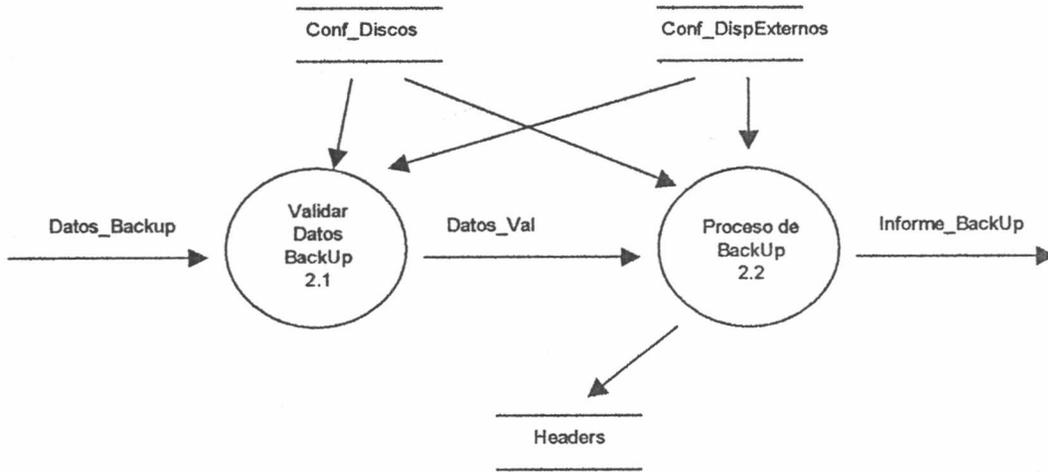
ANEXOS

1. Análisis funcional del BCR

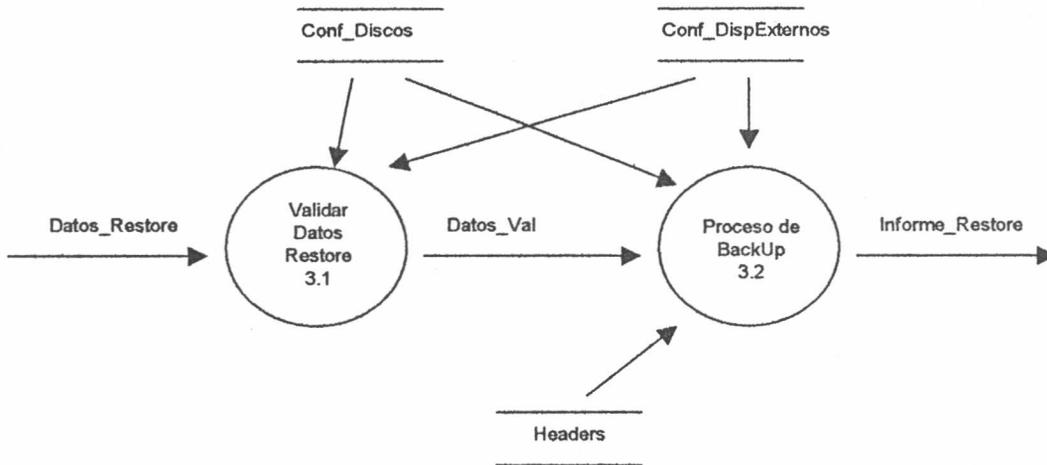
• Nivel 1



2) Backup



3) Restore



2. Descripción de Procesos

1) Establecer Configuración

1. Dar de alta la geometría de los discos
 - Ingresar el Numero de Disco segun el BIOS
 - Ingresar la Descripción del Disco
 - Ingresar la cantidad de Heads
 - Ingresar la cantidad de Cilindros
 - Ingresar la cantidad de Sectores
2. Validar los datos ingresados con el BIOS.
3. Dar de Alta la configuración del dispositivo externo.

2) BackUp

1. Elegir los discos de los que se van a realizar el BackUp
2. Elegir en el dispositivo en el que se va a realizar el BackUp
3. Para cada disco seleccionado
 - 3.1 Guardar en el header ubicado en el dispositivo externo:
 - El Nro de disco que estamos haciendo BackUp
 - El numero de secuencia.
 - La fecha y hora del backup.
 - 3.2 Mientras el dispositivo externo no este completo
 - Leer (cilindro ,head, sector) del disco
 - SI el sector esta dañado
 - Leer (cilindro ,head, sector) del disco
 - SI el sector esta dañado
 - Reportar error.
 - Grabar en el dispositivos externo.
 - 3.3 Si la unidad del dispositivo externo esta completo marcar en el header que continua el backup en otra unidad. pedir otro dispositivos externos y realizar el paso 3.1

3) Restore

1. Elegir los discos de los que se van a realizar el restore.
2. Elegir en el dispositivo desde el cual se va a realizar el Restore.
3. Para cada disco seleccionado
 - 3.1 Leer el header de la unidad del dispositivo externo:
 - El Nro de disco que estamos haciendo BackUp
 - El numero de secuencia.
 - La fecha y hora del backup.
 - 3.2 Si no coincide el disco seleccionado con los datos del header
 - Reportar error
 - Pedir otra unidad
 - 3.3 Mientras el header no reporte fin de secuencia para el disco
 - Leer desde la unidad del dispositivo externo
 - Grabar (cilindro ,head, sector) en el disco
 - SI el sector esta dañado
 - Grabar (cilindro ,head, sector) en el disco
 - SI el sector esta dañado
 - Reportar error y pedir confirmación de continuación
 - 3.4 Si se leyeron todos los datos de la unidad del dispositivo externo leer del header el numero de secuencia de la siguiente unidad y realizar el paso 3.3

3. Escenarios de pruebas

Se realizaron pruebas sobre los siguientes escenario dando como resultado la restauración de la información en su totalidad.

Por razones de no disponer todos los medios de hardware se realizaron con los medios que estuvieron a nuestro alcance la mayor cantidad de escenarios posibles para las verificaciones o pruebas del "BCR", a continuación describiremos algunos de ellos:

Server	Sistema Operativo y Aplicaciones	Discos	Dispositivo Externo	Operación	Tiempo de Operación
Memoria: 64 MB Procesador: Intel Celeron 300	DOS	IDE 42 MB Segate ST 351A/X	Maxtor 82187 A5 2192 MB (Disco IDE)	Backup	2' 40"
				Restore	3' 45"
Memoria: 64 MB	DOS	IDE 42 MB	ZIP Interno Iomega IDE	Backup	4' 30"

Procesador: Intel Celeron 300		Segate ST 351A/X	Iomega IDE de 100 MB	Restore	5' 15"
Server	Sistema Operativo y Aplicaciones	Discos	Dispositivo Externo	Operación	Tiempo de Operación
Memoria: 64 MB Procesador: Intel Celeron 300	DOS	IDE 42 MB Segate ST 351A/X	JAZ Externo SCSI de 2GB	Backup	2' 25"
				Restore	3' 45"
Memoria: 64 MB Procesador: Intel Celeron 300	Novell 3.12	IDE Fujitsu MPC 3032AT 3,24GB	Maxtor 82187 A5 2192 MB (Disco IDE)	Backup (42MB)	2' 16"
				Restore (42MB)	3' 04"
Memoria: 64 MB Procesador: Intel Celeron 300	Novell 3.12	IDE Fujitsu MPC 3032AT 3,24GB	ZIP Interno Iomega IDE de 100 MB	Backup (42MB)	4' 01"
				Restore (42MB)	4' 54"
Memoria: 64 MB Procesador: Intel Celeron 300	Novell 3.12	IDE Fujitsu MPC 3032AT 3,24GB	JAZ Externo SCSI de 2GB	Backup (42MB)	2' 21"
				Restore (42MB)	3' 09"
Memoria: 64 MB Procesador: Intel Celeron 300	Windows 2000 con SQL Server2000	SCSI Segate ST15230 WC 4,32GB	Maxtor 82187 A5 2192 MB (Disco IDE)	Backup (42MB)	0' 57"
				Restore (42MB)	1' 10"
Memoria: 64 MB Procesador: Intel Celeron 300	Windows 2000 con SQL Server2000	SCSI Segate ST15230 WC 4,32GB	ZIP Interno Iomega IDE de 100 MB	Backup (42MB)	2' 50"
				Restore (42MB)	3' 44"
Memoria: 64 MB Procesador: Intel Celeron 300	Windows 2000 con SQL Server2000	SCSI Segate ST15230 WC 4,32GB	JAZ Externo SCSI de 2GB	Backup (42MB)	0' 52"
				Restore (42MB)	1' 03"

4. Instalación y Manual del Usuario

Instalación y Manual del Usuario

INDICE

MANUAL DEL USUARIO.....	1
INDICE.....	2
MENÚ PRINCIPAL.....	3
1 CONFIGURACIÓN	3
1 Discos.....	4
2 ZIP Drive.....	6
2 BACKUP.....	6
3 RESTORE.....	8
4 TABLA DE MENSAJES.....	10

Menú Principal

El Menú Principal consta de cuatro opciones que son las tareas que realiza el operador que es la de configuración del server y luego la realización de backups y restore según la frecuencia deseada y la opción de abandonar el sistema.

MENU PRINCIPAL

1 Configuracion
2 Backup
3 Restore
4 Salir

Ingrese la opción:

A continuación detallamos las cuatro opciones del Menú Principal.

1 Configuración

Esta opción es utilizada para ingresar todos los parámetros necesarios para realizar las operaciones de backup y restore como ser alta/baja de discos y las configuraciones de los mismos. Luego también se define el drive que ocupa el zip drive para realizar los backups o de donde se tomará la información para realizar la restauración.

CONFIGURACION

1 Discos
2 ZIP Drive
3 Salir

Ingrese la opción:

1 Discos

En esta opción da la posibilidad de dar de Alta/Baja los disco que tiene el server al cual posteriormente se realizará el backup.

CONFIGURACION DE DISCOS

1 Alta Disco
2 Baja Disco
3 Salir

Ingrese la opción:

1 ALTA DISCO

En esta opción se ingresan todos los atributos o parámetros necesarios para definir el tipo de disco y su geometría para poder tener el acceso a todo el disco.

CONFIGURACION	ALTA DE DISCOS
Numero de Disco [XX]	: 1
Descripción	: 42 MB
Head [XX]	: 17
Cilindros [XXXX]	: 89
Sectores [XX]	: 456
Confirma lo ingresado? (S) confirma (C) cancela :	

Numero de Disco [XX] : El número de disco es el orden en el cual estan configurados por el BIOS de la maquina o por el BIOS de la controladora. El rango de

este campo es de 0..10 Ej: Número de Disco=0 significa el primer disco del server, Número de Disco=1 el segundo disco, etc..

Descripción: En este campo es aconsejable poner una descripción representativa del tipo de disco al cual pertenece esta configuración, la cantidad máxima de caracteres que se pueden ingresar son 29.

Head: Es la cantidad de heads que tiene el disco.

Cilindros: Cantidad de cilindros del disco.

Sectores : Cantidad de sectores del disco.

Después de ingresar toda la información está la opción de confirmar lo ingresado o la de cancelar para su posterior ingreso o corrección.

2 BAJA DISCO

En esta opción se elige (N) el disco al cual queremos dar de baja (S) porque no pertenece más al server en cuestión o también cuando se da de alta un disco cuya configuración no corresponde.

CONFIGURACION	BAJA DE DISCOS
Numero de Disco [XX]	:1
Descripción	:42 MB
Head [XX]	:17
Cilindros [XXXX]	:89
Sectores [XX]	:456
Para borrar (S) confirma, (C) cancela, (N) siguiente :	

Nota: La baja siempre es lógica.

3 SALIR

Vuelve al menú anterior (Configuración)

2 ZIP Drive

En esta opción se realiza el alta del drive donde se realizará el Backup, el ingreso es un solo caracter el que especificará el drive.

Asigne el drive de ZIP

Drive del ZIP [X] : F

2 Backup

Al elegir la opción de backup se listarán todos los discos que fueron dados de alta anteriormente con su número de disco y su respectiva descripción, para elegir los discos a ser backapeados hay que digitar el número de disco. Después de realizar la elección de todos los discos con 99 se da por finalizada la elección.

BACKUP

Numero de Disco	Descripción
1	42 MB

Ingrese discos a resguardar (Nº de disco) : 1

A continuación se desplegará el Drive seleccionado donde se almacenará el backup, la lista de discos que serán resguardados en el drive y queda a la espera de la confirmación para realizar el backup.

DRIVE DONDE SE REALIZARA EL BACKUP

Drive del ZIP: E

DISCOS ELEGIDOS A SER RESGUARDADOS

Numero de Disco	Descripción
1	42 MB

Realizar el Backup (B) Cancelar (C) :

Si se confirma el backup inmediatamente empezará a realizar el backup en el orden ascendente de la numeración de los discos.

DRIVE DONDE SE REALIZARA EL BACKUP

Drive del ZIP: E

DISCOS ELEGIDOS A SER RESGUARDADOS

Numero de Disco	Descripción
1	42 MB

Realizar el Backup (B) Cancelar (C) : 0 % +

A medida que realiza el backup mostrará un porcentaje del backup que ha realizado para mostrar que el programa está trabajando, a continuación muestra un conjunto de flechas que giran como muestra de actividad.

3 Restore

Al elegir la opción de restore se listarán todos los discos que fueron dados de alta anteriormente con su número de disco y su respectiva descripción, se elegirán los discos a ser restaurados por el número de disco, después de realizar la elección de todos los discos para dar por terminada la operación se digitará 99 dando por finalizada la elección.

RESTORE	
Numero de Disco	Descripción
1	42 MB

Ingrese discos a restaurar (Nº de disco) : 1

A continuación desplegará el Drive seleccionado de donde se tomará el backup, la lista de discos que serán restaurados y queda a la espera de la confirmación para realizar el restore.

DRIVE DONDE SE REALIZARA EL RESTORE	
Drive del ZIP: E	
DISCOS ELEGIDOS A SER RESTAURADOS	
Numero de Disco	Descripción
1	42 MB

Realizar la Restauración (R) Cancelar (C) :

Si se confirma el restore inmediatamente empezará a realizarse en el orden ascendente de la numeración de los discos.

DRIVE DE DONDE SE REALIZARA EL RESTORE	
Drive del ZIP: E	

DISCOS ELEGIDOS A SER RESTAURADOS	
Numero de Disco	Descripción

1	42 MB
Realizar la Restauración (R) Cancelar (C) : 0 % +	

A medida que realiza el restore mostrará un porcentaje del restore que ha realizado para mostrar que el programa está trabajando, a continuación muestra un conjunto de flechas que giran como muestra de actividad.

4 Tabla de Mensajes

Mensaje 1 : "Error al abrir el archivo [Nombre del Archivo]"

Descripción: Este error se produce cuando se quiere abrir un archivo y este no se encuentra generado o no se encuentra el disquete de configuración en el drive A o cuando no se encuentra el ZIP en el drive seleccionado para realizar el backup.

Mensaje 2 : "Error al crear el archivo [Nombre del Archivo]"

Descripción: Este error se produce cuando se quiere crear un archivo y no se puede generar el mismo, esto se puede dar porque no se encuentra el disquete de configuración en el drive A o cuando no se encuentra el ZIP en el drive seleccionado para realizar el backup.

Mensaje 3 : "Nro. de disco existente"

Descripción: Este mensaje se genera porque se quiere dar de alta un disco que ya fue dado de alta.

Mensaje 4: "Esta elegido"

Descripción: Este mensaje se presenta cuando el nro. de disco a realizar el restore o el backup ya fue seleccionado.

Mensaje 5: "No esta dado de alta"

Descripción: Este mensaje hace referencia cuando se ingresa un nro. de disco a ser restaurado pero no se encuentra en el archivo de configuración.

Mensaje 6: "Este no contiene el Backup del disco"

Descripción: Este mensaje es cuando el ZIP introducido no contiene el backup del nro. de disco que se quiere restaurar.

Mensaje 7: "El drive no fue seleccionado"

Descripción: Cuando se pide realizar y el drive de donde se tomar el backup o restore no fue configurado adecuadamente.

Mensaje 8: "No fue elegido ningún disco para realizar su backup"

Descripción: No se ingreso ningún nro. de disco para ser backapeado.

5. Bibliografía

- [AdEz] **Adaptec EZ-SCSI User's Guide.**
- [Ad91] **Addison-Wesley Publishing Company Inc.,** Phoenix; Technical Reference Series; System Bios for IBM Pcs, Compatibles and EISA Computers; 1991
- [Ad92] **Advanced Micro Devices,** Personal Computer Microprocesors; Data Books; 1992.
- [Ad98] **Administration of SQL Server,** Microsoft Corp; 1998
- [An78] **Andrews S. Tanenbaum,** Sistemas Operativos: Diseño e Implementación, Prentice-Hall Hispanoamérica; 1988.
- [An88] **Andrew S. Tanenbaum,** "Operating Systems: Design and Implementation", 1988.
- [Bo83] **Boira, Jorge,** "Construcción de sistemas operativos", Edit. Kapelusz (IV EBAI)
- [Br78] **Brian W. Kernighan, Dennis M. Ritchie,** El Lenguaje de Programación C, Prentice-Hall Hispanoamerica, 1978.
- [CBpG] **Colorado Backup - User's Guide.**
- [DU99] **Duncan;** ROM BIOS IBM Referencia, Editorial Anaya 1999
- [Fci88] **Flynn, Michel J., Mitchell, Chad L. & Mulder Johannes M.;** " And now a case for more complex instruction sets"; IEEE Computer; Vol 20, Nro. 9.
- [FuBo] **Furth, Borivoje;** IEEE Computer; "A definition of complex instruction set computer (CISC) architectures"; Vol. 20, Nro 9.
- [HaJo] **Hayes, John P.;** "Computer architecture & organization", Edit. McGraw-Hill.
- [Ha87] **Harvey M. Deitel,** "An Introduction to Operating Systems", Addison – Wesley Publishing Company., 1987.
- [InII] **Intel** Microprocesor and Periferical HandBook; Volumen I y II.
- [In90] **Intel;** 386 DX Microprocesador; Programer's Reference Manual, 1990.
- [Jo87] **John H. Crawford,** Patrick P.Gelsiner; Programing the 80386; Sybex, 1987.
- [MaBa] **Maurice J. Bach,** The Design of the Unix Operating System, Prentice-Hall EngleWood Cliffs.
- [Mi88] **Milan Milenkovic,** "Operating Systems: Concepts and Design", 1988.
- [NCR89] **NCR System 3000 System Administrator Guide.**
- [Pe85] **Peter Norton,** "Guia del programador",1985.

- [Pe87] **J. Peterson A. Silberschatz**; Operating Systems Concepts, Addison Wesley, 1987.
- [Ph91] **Phonix-Technical Reference Series**
System BIOS for IBM Pcs, Compatibles and EISA Computers Addison-Wesley
Publishing Company Inc, 1991.
- [Te72] **Teorey, T.J., y Pinkerton, T.B.**, "A comparative Analysis of Disk Scheduling
Policies", CACM Vol. 15, num 3, 1972, pag. 177-84.
- [ToHe] **Tomas E Anderson**, Henry M. Levy, Brian N. Bershad, D. Lazonwska; The
Interaction of Architecture and Operating System Desig, Departament of Computer
Science and Engineering University of Washinton.
- [SR00] **Smith Roderick W.** ; The multi-boot configuration Handbook, 2000.
- [SW01] **Stallings Williams**; Operating Systems, 2001
- [Uns92] **UNIX SVR4 MP System Files and Devices** Reference Manual April 1992 D2-
0373-B. Aspray, William; "The Stored program concept", IEEE Spectrum, Vol. 27, Nro
9.
- [VIFR] **Vicent F. Russo, Peter W. Mandany, Roy H. Cambell.**; C++ and Operating
Systems Performance: A case Study. University of Illinois at Urbana-Champaingn,
Department of Computer Science.
- [VMSP] **VMSP** - Operator's Guide.

6. Código Fuente

Ver soporte magnético.