RAID

31 Marzo 2011

1. Generalidades

Un RAID -del inglés, *Redundant Array of Independent Disks* (Arreglo Redundante de Discos Independientes)[1]- es un dispositivo de almacenamiento constituido por un conjunto de dos o más discos rígidos que permiten almacenar información e incluso replicarla, de modo que ante una falla exista la posibilidad de recuperar los datos. La forma en que los datos se distribuyen entre los diferentes discos, y el grado de *redundancia* proporcionada depende de la configuración particular del RAID. La redundancia consiste en mantener copia de los datos en más de un disco. En general, los niveles de RAID que proveen redundancia permiten que al menos un disco falle sin provocar la pérdida de los datos en él almacenados. Esta característica posibilita que el RAID continúe funcionando en caso de un fallo en el disco, porque los datos almacenados siguen disponibles para que el usuario los acceda.

El nivel más simple de RAID agrupa varios discos rígidos -en general, de igual capacidad- en una única unidad lógica, de manera tal que el sistema operativo y el usuario los ven como uno solo. Su principal objetivo es proveer mejoras en los siguientes aspectos:

1. **Tolerancia a fallos.** Se refiere, a grandes rasgos, a la capacidad del sistema de detectar fallos y recuperarse ante su ocurrencia.

2. **Rendimiento (throughput).** Está asociado a la velocidad de lectura y escritura del dispositivo en su conjunto.

3. **Capacidad.** Se refiere a la capacidad total del dispositivo. Generalmente, un dispositivo RAID compensa en conjunto la capacidad reducida de cada disco individual.

Así es que la tecnología RAID distribuye la tarea de un disco entre varios, en busca de mejorar el rendimiento, la capacidad y la seguridad del sistema. La *tolerancia a fallos* está íntimamente ligada al concepto de *seguridad*, debido a que si los datos del sistema pueden ser accedidos aun habiendo ocurrido una falla, se dice que el sistema provee *disponibilidad*: los datos pueden ser accedidos, y con la menor cantidad de dificultades posible.

La distribución de datos en varios discos puede gestionarse por hardware dedicado o por software. Además existen sistemas RAID híbridos basados en software y hardware específicos.

En la implementación por **software** el sistema operativo gestiona los discos del arreglo a través de una controladora de disco común (IDE/ATA, Serial ATA, SCSI). Está alternativa ha sido considerada tradicionalmente una solución más lenta, pero con el rendimiento de las CPUs modernas puede llegar a ser más rápida que algunas implementaciones hardware, a expensas de restar tiempo de proceso al resto de las tareas del sistema.

Una implementación de RAID basada en **hardware** requiere al menos una controladora RAID específica -ya sea como una tarjeta de expansión independiente o integrada en la placa base- que gestione los discos y efectúe los cálculos de paridad. Esta opción suele ofrecer un mejor rendimiento y hace que el soporte del sistema operativo sea más sencillo. Estas implementaciones suelen soportar

[1] Su nombre fue redefinido por la industria, dado que inicialmente la letra *I* provenía de *Inexpensive*, es decir, *Baratos*.

sustitución en caliente (*hot swapping*) permitiendo que los discos que fallan puedan ser reemplazados sin necesidad de detener el sistema.

Los RAIDs híbridos se han popularizado con la introducción de controladoras RAID hardware baratas. El hardware es una controladora de disco normal sin características RAID, pero el sistema incorpora una aplicación de bajo nivel que permite a los usuarios construir RAIDs controlados por la BIOS. En realidad, estos sistemas efectúan todos los cálculos por software (los realiza la CPU), con la consiguiente pérdida de rendimiento, y están restringidos a una única controladora de disco.

Todas las implementaciones pueden soportar el uso de uno o más discos de reserva que puede/n usarse inmediatamente tras el fallo de un disco del RAID.

2. Niveles de RAID

Diferentes configuraciones de RAID dan lugar a distintos *niveles de RAID*[2]. A este punto, cabe aclarar que el presente material sólo describe los niveles de RAID del 0 al 6. Queda en manos del lector el estudio de las distintas implementaciones que con el tiempo han ido surgiendo en base a las que aquí se describen.

Generalizando, todas las configuraciones RAID hacen uso de algunas de las siguientes estrategias:

1. *Striping (escritura en franjas):* Consiste en segmentar los archivos en bloques o sectores y organizarlos en *stripes* (bandas o franjas) de modo de almacenar los distintos bloques del stripe en múltiples discos simultáneamente a través de distintos canales de transferencia, paralelizando las lecturas/escrituras e incrementando así la velocidad con que se efectúan ambas operaciones.

2. *Mirroring:* Es la creación de copia/s idéntica/s (copia/s espejo) de la información en uno o más discos. Esto permite leer varios sectores de datos de cada disco a la vez por medio de canales de transferencia de datos distintos, y a la vez favorece los aspectos involucrados en la seguridad de los datos, dado que ante una falla en un disco, los datos podrian accederse desde otro disco que almacene la copia.

2.1. RAID 0

Conocido como *Striped Disk Array* o *Conjunto Dividido*, debido a que almacena los bloques de cada archivo equitativamente entre los diferentes discos. Sin embargo, no provee redundancia de datos, y es por ésto comúnmente cuestionado como RAID. La **Figura 1** provee un esquema de un RAID 0 sencillo y la distribución de datos en el mismo. Se necesitan dos unidades de disco como mínimo para implementar un RAID de este nivel.

Sea un RAID 0 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_0 = N * C$; C_0 : capacidad del RAID 0,
- $R_0 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-N*C}{N*C} = 0 \Rightarrow no\,provee\,tolerancia\,a\,fallos$[3] ; R_0 :redundancia del RAID 0,
- $V_0 = N * V$; V_0 : velocidad de transferencia (para lecturas y escrituras) del RAID 0, limitada por la velocidad de la controladora RAID.

Los RAID 0 pueden crearse con discos de diferentes tamaños, en cuyo caso se limita su capacidad a la del disco más pequeño. Por ejemplo, si se tiene un disco de 120 Gb y otro de 100 Gb, C_0 será igual a 200 Gb.

[2] El término *nivel* no es quizás el más apropiado, porque no se trata de una jerarquía sino de diferentes organizaciones de los datos.

[3] Se considera que el *Espacio Real* es la suma de las capacidades de cada uno de los discos. Mientras que el *Espacio Aprovechado* se refiere al espacio destinado a almacenar información, sin redundancia alguna; justamente, el *Espacio Aprovechado* coincide con la capacidad del RAID.

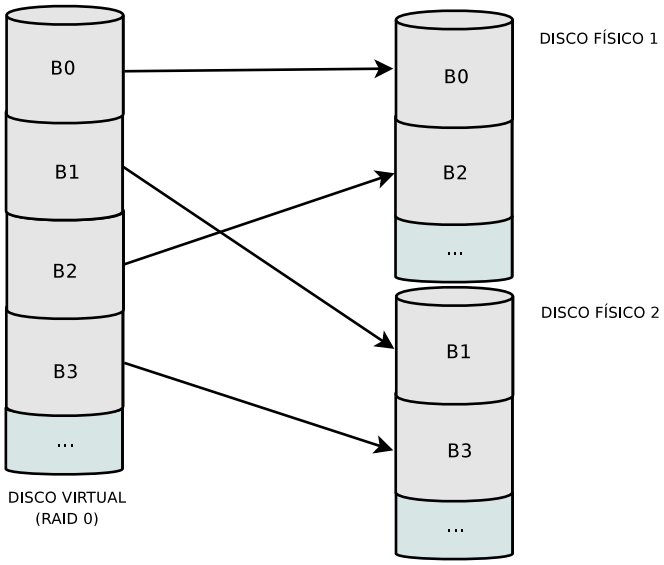

Figura 1: Esquema de una configuración RAID 0 con N=2 discos. B_i : *bloque de datos* , $0 \leq i \leq 3$.

Entonces por un lado, provoca una mejora en el rendimiento general debido a que se incrementan en promedio N veces la velocidad de lectura y de escritura, pues es posible efectuar tales operaciones en simultáneo sobre todos los discos. Si todos los sectores accedidos están en el mismo disco, el tiempo de búsqueda será el de dicho disco. Si los sectores están distribuidos equitativamente entre los discos, el tiempo de búsqueda estará entre el más rápido y el más lento de los discos del conjunto.

Pero por otro lado, no es tolerante a fallos debido a la ausencia de redundancia: si se daña un disco o parte de él, se pierde parte de los datos de cada uno de los archivos, y no hay manera alguna de recuperarla a partir de los discos restantes. La probabilidad de fallo del RAID 0 es igual a la suma de las probabilidades de cada disco, por tanto su confiabilidad reduce cuanto más crece N.

El RAID 0 es entonces útil en sistemas en los que las pérdidas de datos no representen un problema grave y/o en sistemas en los que es altamente deseable un buen rendimiento como, por ejemplo, sistemas destinados a juegos. También es una solución útil en sistemas sobre los que se realizan operaciones secuenciales con archivos de gran tamaño como ser aquellos sobre los que se ejecutan aplicaciones de edición de video, imágenes o audio. Otra utilidad, hoy en día no tan relevante, es para aumentar la capacidad de almacenamiento del sistema ante sistemas operativos que establezcan un límite máximo sobre el número de unidades lógicas[4].

2.2. RAID 1

También llamado *Mirroring and Duplexing* o *Conjunto Espejo*. Utiliza un disco para el almacenamiento de los datos del sistema, y en los discos restantes del arreglo guarda copias espejo del primero. La **Figura 2** provee un esquema de un RAID 1 básico y la distribución de datos en el mismo. Cabe aclarar que hay que contar con al menos dos unidades de disco para implementar un RAID 1.

Sea un RAID 1 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_1 = C$; C_1 : capacidad del RAID 1,
- $R_1 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-C}{N*C} = (\frac{N-1}{N})$; R_1 : redundancia del RAID 1,

[4]Los sistemas Windows en general utilizan letras para montar las distintas unidades.

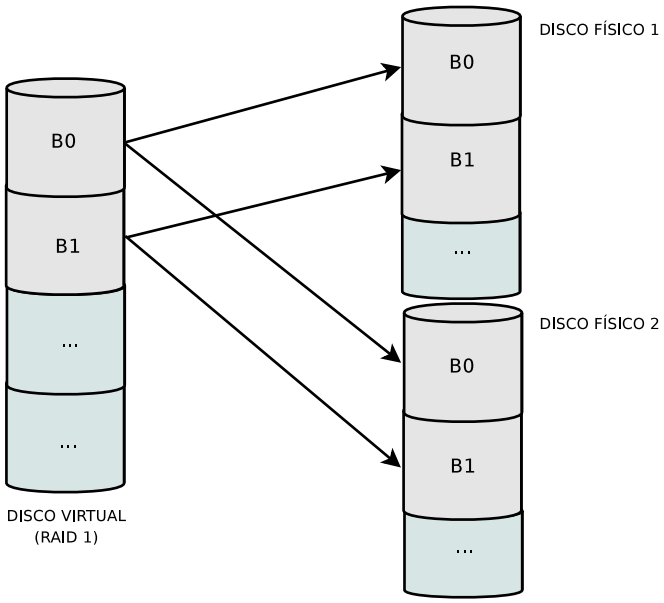

Figura 2: Esquema de una configuración RAID 1 con N=2 discos. B_i : *bloque de datos* , $0 \leq i \leq 1$.

- $V_1 = N * V$; V_1 : velocidad de transferencia para lecturas del RAID 1 *(la velocidad de escritura es aproximadamente la de un disco, o sea, V).*

Por un lado, el hecho de que todos los datos están almacenados por igual en distintos discos provoca una ganancia en la velocidad de lectura, puesto que pueden leerse en paralelo[5]. Por el otro, siendo que los datos deben ser escritos en cada uno de los discos del RAID 1, la velocidad de escritura no se ve favorecida respecto a un único disco.

Para que el RAID en su totalidad falle, es preciso que todos y cada uno de sus discos fallen -debido a que define un disco de datos y los restantes son copias idénticas del mismo-, por tanto, la probabilidad de fallo del conjunto es igual al producto de las probabilidades de fallo de cada disco por separado. Así es que esta configuración es altamente tolerante a fallos.

Resulta útil cuando la seguridad de los datos es lo primordial en el sistema, más aun si las consultas son frecuentes, por lo que es deseable una buena performance en lectura. Esta configuración claramente no favorece la capacidad de almacenamiento, puesto que destina la mayor cantidad del espacio disponible a la redundancia.Por lo tanto, se convierte en una alternativa costosa para soportar grandes sistemas -pues si se quisiera ganar capacidad, habría que agregar discos, pero de a pares: uno para almacenamiento y otro para redundancia-. En consecuencia, RAID 1 constituye una buena alternativa para servidores de archivos pequeños.

2.3. RAID 2

Este nivel de RAID segmenta los datos a nivel de bits en lugar de bloques, y utiliza *código de Hamming*[6] para proveer *detección y corrección de errores.* La **Figura 3** provee un esquema de un RAID 2 simple y la distribución de datos en el mismo.

Si el RAID 2 segmenta los datos en palabras de 32 bits, se requieren 6 bits de paridad para formar una palabra Hamming de 38 bits, y 1 bit más de paridad de la palabra completa ($2^6 > 32 + 6 + 1$);

[5]Para maximizar el rendimiento de un RAID 1 es recomendable utilizar controladores de disco independientes para cada disco (*splitting* o *duplexing*).

[6]Código corrector de errores, desarrollado por R.W. Hamming en 1950, se basa en los conceptos de bits redundantes y *Distancia Hamming.* La Distancia Hamming H entre dos secuencias de bits *S1* y *S2* de igual longitud está dada por el número de bits en que difieren.

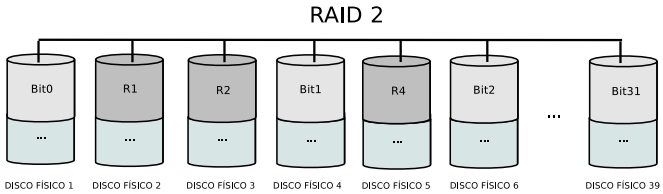

Figura 3: Esquema de una configuración RAID 2 con N=39 discos. R_1, R_2, R_4 : *bits redundancia.*

estos 7 bits extra constituyen los bits de redundancia que se intercalan entre los discos cubriendo las posiciones potencia de 2: R1, R2, R4, R8, etc. Luego, si cada palabra se distribuye uniformemente a nivel de bits, teóricamente se necesitan al menos 39 discos para su implementación y aún así no se alcanza un nivel significativo de redundancia, tal como muestra la segunda ecuación abajo. A su vez, requiere que los discos estén perfectamente sincronizados por la controladora para funcionar al unísono.

Para un RAID 2 como el de la **Figura 3** conformado por N=39 discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_2 = (N - r) * C = 32 * C$; C_2 : capacidad del RAID 2 y r : cantidad de bits de redundancia (7 en este caso),
- $R_2 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-(N-r)*C}{N*C} = \frac{r}{N} \approx 0,18$; R_2 : redundancia del RAID 2,
- $V_2 = (N - r) * V$; V_2 : velocidad de transferencia del RAID 2.

Pese a permitir tasas de trasferencia muy altas y a que la potencial pérdida de un disco no causaría problemas (porque sólo implicaría perder 1 bit de cada palabra, lo cual el código de Hamming puede resolver al vuelo), esta configuración actualmente no se usa. Por un lado porque el número de solicitudes de E/S por segundo que pueden atender no mejora respecto a un único disco, porque como cada bloque de datos se encuentra dividido entre todos los discos del arreglo, cualquier operación de lectura o escritura de un bloque exige acceder sincrónicamente a todos los discos impidiendo el acceso concurrente a cualquier otro bloque. Por otro lado, su implementación cobra sentido con una gran cantidad de unidades de disco, exactamente sincronizadas y de características especiales. Además, ha sido pensado para ser implementado con discos que carecen de mecanismos de detección y corrección de errores interna, y en la actualidad todos los discos SCSI los proveen.

Como en esencia es una tecnología de acceso paralelo, RAID 2 resulta apropiado para aplicaciones que requieran una alta tasa de transferencia, y no así para aquellas que requieran una alta tasa de demanda de E/S.

2.4. RAID 3

Conocido también como *Bit Interleaved Parity* o *Bit de Paridad Intercalado*, este nivel es una versión simplificada del nivel anterior, que segmenta los datos a nivel de byte -ya no de bit- y calcula información de paridad por cada palabra, almacenándola en un disco de paridad dedicado. Tal como en un RAID 2, requiere que las unidades de disco estén perfectamente sincronizadas debido a que cada palabra de datos se encuentra distribuida entre múltiples discos. La **Figura 4** provee un esquema de un RAID 3 sencillo y la distribución de datos en él. Su implementación requiere como mínimo tres unidades de disco.

Sea un RAID 3 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_3 = (N - 1) * C$; C_3 : capacidad del RAID 3,
- $R_3 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-(N-1)*C}{N*C} = \frac{1}{N}$; R_3 : redundancia del RAID 3,

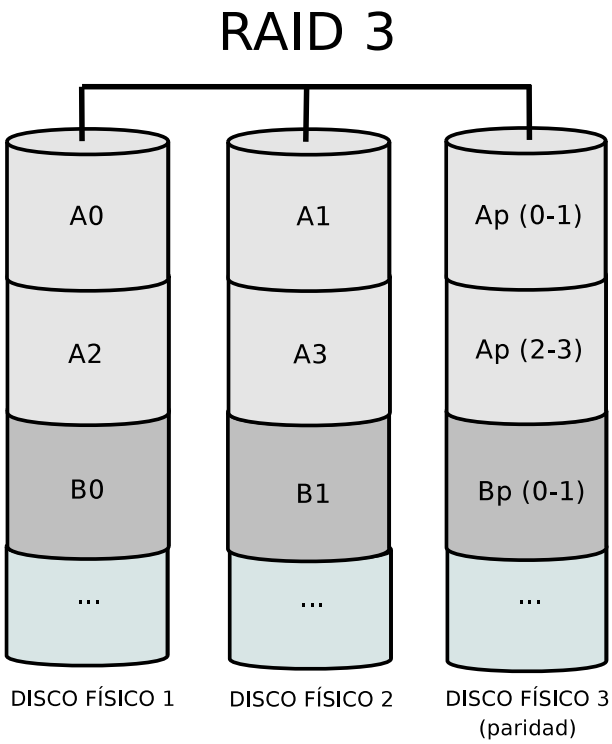

Figura 4: Esquema de una configuración RAID 3 con N=3 discos. A_i, B_i : *byte de datos*.

- $V_3 = (N-1) * V$; V_3 : velocidad de transferencia del RAID 3.

La información de paridad se calcula aplicando la operación lógica XOR -cuyo símbolo es $\oplus$- entre los *bytes de datos* escritos en los discos restantes. La **Figura 5** muestra el comportamiento de dicha función booleana por medio de su tabla de verdad.

A	B	$A \oplus B$
0	*0*	*0*
0	*1*	*1*
1	*0*	1
1	1	0

Figura 5: Tabla de verdad de la función booleana XOR.

Se utiliza en particular XOR porque la misma posibilita la recuperación de los datos dañados en base a los sanos y a la paridad. Es sencillo comprobar que si $A \oplus B = C \Rightarrow C \oplus A = B$ *y* $C \oplus B = A$.

Ejemplo:

Sea un RAID 3 como el de la **Figura 4**, *se asume que se almacenan los bytes de datos* A_0 *y* A_1 *en los discos 1 y 2 respectivamente, por tanto en el disco de paridad (disco 3) se almacena:* $A_p(0-1) = A_0 \oplus A_1$. *Luego, si se dañara por ejemplo el disco 2, el byte* A_1 *podría recuperarse aplicando la misma función como sigue:* $A_1 = A_p \oplus A_0$.

Este nivel tolera la falla de un único disco a la vez sin ocasionar pérdidas de datos, dado que los datos en el disco averiado pueden reconstruirse en un disco de recambio por medio de la información registrada en los otros discos.

Debido a que escribe los datos en grandes bloques de información es una alternativa apropiada para aplicaciones en que se realizan operaciones secuenciales con archivos de gran tamaño, como editores de video o imágenes.

Si bien ofrece muy altas tasas de transferencia no es habitualmente usado en la práctica, fundamentalmente porque, al igual que el RAID 2, no puede atender varias solicitudes de E/S simultáneas. En el ejemplo de la **Figura 4**, una petición del bloque *A* formado por los bytes *A0* a *A3* requiere de los dos discos de datos para atender la solicitud. Entonces, cualquier solicitud simultánea de otro bloque, por ejemplo el *B*, debe esperar a que la primera concluya.

2.5. RAID 4

Conocido también como *Dedicated Parity Drive,* o bien, *Unidad de Paridad Dedicada.* Segmenta los datos a nivel de bloque, y distribuye los bloques de cada archivo equitativamente entre los discos que lo conforman -hasta aquí se asemeja a un RAID 0-, a excepción de un disco que destina para almacenar la información de paridad calculada entre los bloques de datos escritos. La **Figura *6*** esquematiza un RAID 4 sencillo y la distribución de datos en el mismo.

Lo anterior trae aparejadas dos condiciones:

- **condición 1** - es menester contar con *al menos 3 discos* para crear una unidad RAID de nivel 4 y,
- **condición 2** - el *disco dedicado a la paridad* debe ser *N veces más rápido* que los restantes a los efectos de evitar que se convierta en un cuello de botella, afectando negativamente la velocidad de escritura del RAID.

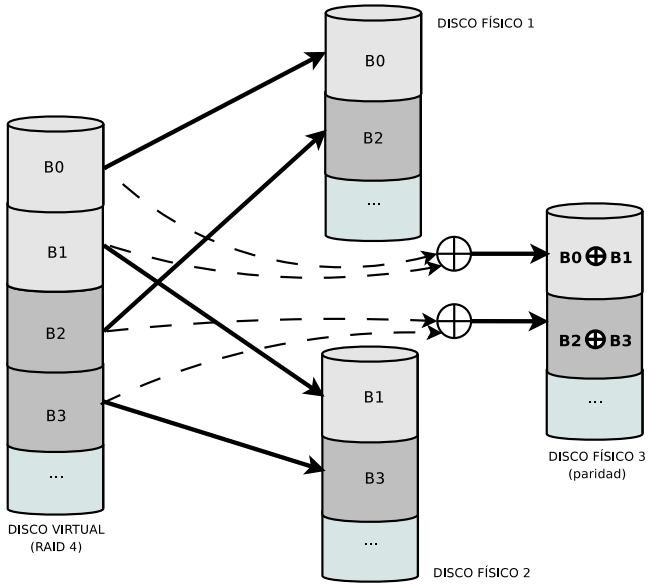

Figura 6: Esquema de una configuración RAID 4 con N=3 discos. B_i : *bloque de datos* , $0 \leq i \leq 3$.

Sea un RAID 4 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_4 = (N-1) * C$; C_4 : capacidad del RAID 4,
- $R_4 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-(N-1)*C}{N*C} = \frac{N*C-(N-1)*C}{N*C} = \frac{1}{N}$; R_4 : redundancia del RAID 4,
- $V_4 = (N-1) * V$; V_4 : velocidad de transferencia para lecturas del RAID 4[7].

[7] En principio podría atender varias peticiones de escritura en simultáneo, a lo sumo N-1, pero para ello es imprescindible que se cumpla la **condición 2**.

La información de paridad se calcula aplicando la operación lógica XOR entre los *bloques de datos* escritos en los discos restantes.

Un dispositivo RAID 4 provee buena tolerancia a fallos y mejoras en el rendimiento respecto de un único disco, aunque ante actualizaciones pequeñas. La mera modificación de un sector requiere que se vuelvan a leer todas las unidades de disco para recalcular la paridad, y que ésta sea reescrita en el disco dedicado a tal propósito.

2.6. RAID 5

Conocido también como *Block Interleaved Distributed Parity* o *Conjunto Dividido con Paridad Distribuida*. Este nivel es similar al RAID 4, pero difiere en cómo guarda la información de paridad. No lo hace en un único disco, sino que la distribuye uniformemente entre los discos del arreglo, por turno circular. De este modo, en un RAID 5 todos los discos almacenan tanto datos como paridad. La **Figura 7** brinda un esquema de un RAID 5 sencillo y la distribución de datos en el mismo.

Al igual que el RAID 4, su implementación requiere al menos tres discos. La falla de un disco no afecta la disponibilidad de los datos, pues puede reconstruirse la información perdida. No obstante, la falla de un segundo disco provoca la pérdida total de los datos. Por otro lado, ya no se establece la restricción de que haya un disco más veloz que el resto, debido a que ya no se tiene un disco dedicado para la paridad, sino que todos desempeñan la misma función.

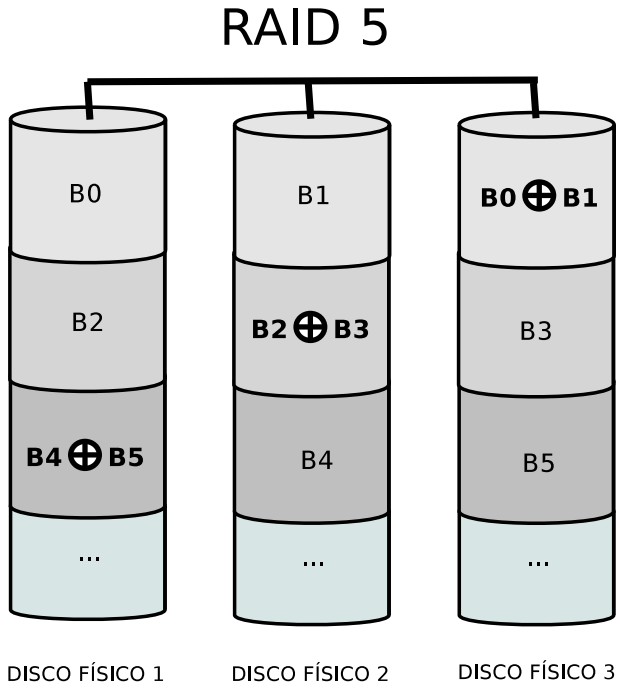

Figura 7: Esquema de una configuración RAID 5 con N=3 discos. B_i : *bloque de datos* , $0 \leq i \leq 5$.

Sea un RAID 5 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_5 = (N-1) * C$; C_5 : capacidad del RAID 5,
- $R_5 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-(N-1)*C}{N*C} = \frac{N*C-(N-1)*C}{N*C} = \frac{1}{N}$; R_5 : redundancia del RAID 5,
- $V_5 = (N-1) * V$; V_5 : velocidad de transferencia del RAID 5.

A mayor número de discos N, mayor probabilidad de que fallen simultáneamente dos discos y mayor tiempo de reconstrucción ante fallas. Provee en general buena tolerancia a fallos y buen rendimiento. Es muy apropiado para múltiples transacciones pequeñas como emails, procesadores de texto, hojas de cálculo y aplicaciones de bases de datos.

2.7. RAID 6

Conocido como *Independent Data Disks with Double Parity* o *Discos de Datos Independientes con Doble Paridad*. Este nivel provee segmentación de los datos a nivel de bloque y paridad distribuida entre todos los discos. Extiende al RAID 5 dado que agrega un segundo bloque de paridad; ambos bloques son almacenados uniformemente entre los discos del arreglo, por turno circular. La **Figura 8** muestra un esquema de un RAID 6 simple y la distribución de datos en él. Se requieren al menos cuatro unidades de disco para implementar un RAID 6.

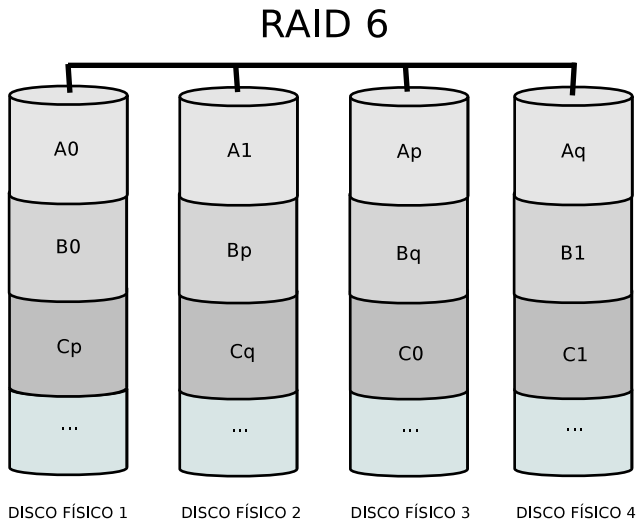

Figura 8: Esquema de una configuración RAID 6 con N=4 discos. A_i, B_i, C_i : *bloque de datos* , $0 \leq i \leq 2$; A_p, B_p : *bloques de paridad*.

Sea un RAID 6 conformado por N discos de capacidad C, con V la velocidad de transferencia de cada disco, se determina:

- $C_6 = (N-2) * C$; C_6 : capacidad del RAID 6,
- $R_6 = \frac{Espacio\,Real - Espacio\,Aprovechado}{Espacio\,Real} = \frac{N*C-(N-2)*C}{N*C} = \frac{2}{N}$; R_6 : redundancia del RAID 6,
- $V_6 = (N-2) * V$; V_6 : velocidad de transferencia para lecturas del RAID 6.

Resulta ineficiente cuando se usa una cantidad de discos N muy pequeña. Conforme crece N, se obtiene un mejor aprovechamiento del espacio, aunque también aumenta la probabilidad de que dos discos fallen simultáneamente. Existen pocos ejemplos comerciales en la actualidad, porque su costo de implementación es mayor desde que requiere controladoras que soporten doble paridad, las cuales son más complejas y costosas.

Ofrece alta tolerancia a fallos, debido al doble nivel de redundancia; puede soportar hasta dos averías de disco sin perder datos -o la falla de un segundo disco cuando se está reconstruyendo otro- y proporciona una reconstrucción más rápida de los datos.

Si bien el rendimiento de las operaciones de lectura se ve favorecido, la tasa de escritura se ve levemente desmejorada debido al proceso de cálculo de paridad. Por tanto, RAID 6 es útil cuando la seguridad prima sobre el rendimiento.

Referencias

[Tan03] **A. Tanembaum.** *Sistemas Operativos Modernos, 2da. Ed.* Pearson Educación, 2003. ISBN: 9702603153.

[WEs09] *"Entrada sobre RAID en línea de Wikipedia en Español"*, Abr 2009 (http://es.wikipedia.org/wiki/RAID)