RAID

Como se dijo anteriormente, el ritmo de mejora de prestaciones en memoria secundaria ha sido considerablemente menor que en procesadores y en memoria principal. Esta desigualdad ha hecho, quizás, del sistema de memoria de disco el principal foco de optimización en las prestaciones de los computadores.

Como en otras áreas de rendimiento de los computadores, los diseñadores de memorias de disco reconocen que si uno de los componentes sólo se puede llevar a un determinado límite, se puede conseguir una ganancia en prestaciones adicional usando varios de esos componentes en paralelo. En el caso de la memoria de disco, esto conduce al desarrollo de con- juntos de discos que operen independientemente y en paralelo. Con varios discos, las peticiones separadas de E/S se pueden gestionar en paralelo, siempre que los datos requeridos residan en discos separados. Además, se puede ejecutar en paralelo una única petición de E/S si el bloque de datos al que se va a acceder está distribuido a lo largo de varios discos.

Con el uso de varios discos, hay una amplia variedad de formas en las que se pueden organizar los datos y en las que se puede añadir redundancia para mejorar la seguridad. Esto podría dificultar el desarrollo de esquemas de bases de datos que se pueden usar en numerosas plataformas y sistemas operativos. Afortunadamente, la industria está de acuerdo con los esquemas estandarizados para el diseño de bases de datos para discos múltiples, conocidos como RAID (Redundant Array of Independent Disks, «conjunto redundante de discos independientes»). El esquema RAID consta de seis niveles independientes, desde cero hasta cinco. Estos niveles no implican una relación jerárquica, sino que designan métodos diferentes que poseen tres características comunes:

1. RAID es un conjunto de unidades fisicas de disco vistas por el sistema operativo co- mo una única unidad lógica.
2. Los datos se distribuyen a través de las unidades físicas del conjunto de unidades.
3. La capacidad de los discos redundantes se usa para almacenar información de paridad que garantice la recuperación de los datos en caso de fallo de disco.

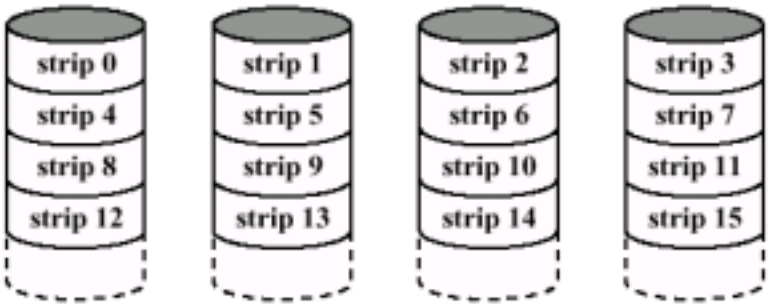
Los detalles de las características segunda y tercera, cambian según los distintos niveles RAID. RAID 0 no soporta la tercera característica.

El término RAID fue originalmente ideado en un artículo de un grupo de investigación de la Universidad de California en Berkeley. El artículo perfilaba varias configuraciones y aplicaciones RAID, e introducía las definiciones de los niveles RAID que todavía se usan. La estrategia RAID reemplaza una unidad de disco de gran capacidad por unidades múltiples de menor capacidad, y distribuye los datos de forma que se puedan habilitar accesos simultáneos a los datos de varias unidades, mejorando, por tanto, las prestaciones de E/S, y permitiendo más fácilmente aumentos en la capacidad.

La única contribución de la propuesta RAID es, efectivamente, hacer hincapié en la necesidad de redundancia. El uso de varios dispositivos, además de permitir que varias cabezas y actuadores operen simultáneamente, consiguiendo mayores velocidades de E/S y de transferencia, incrementa la probabilidad de fallo. Para compensar esta disminución de segurida RAID utiliza la información de paridad almacenada, que permite la recuperación de datos perdidos debido a un fallo de disco.
A continuación examinaremos cada nivel de RAID. De los niveles 2 y 4 no se ofrecen comercialmente y no es probable que consigan aceptación industrial. Sin embargo, la descripción de estos niveles ayuda a clarificar las elecciones de diseño en algunos de los otros niveles.

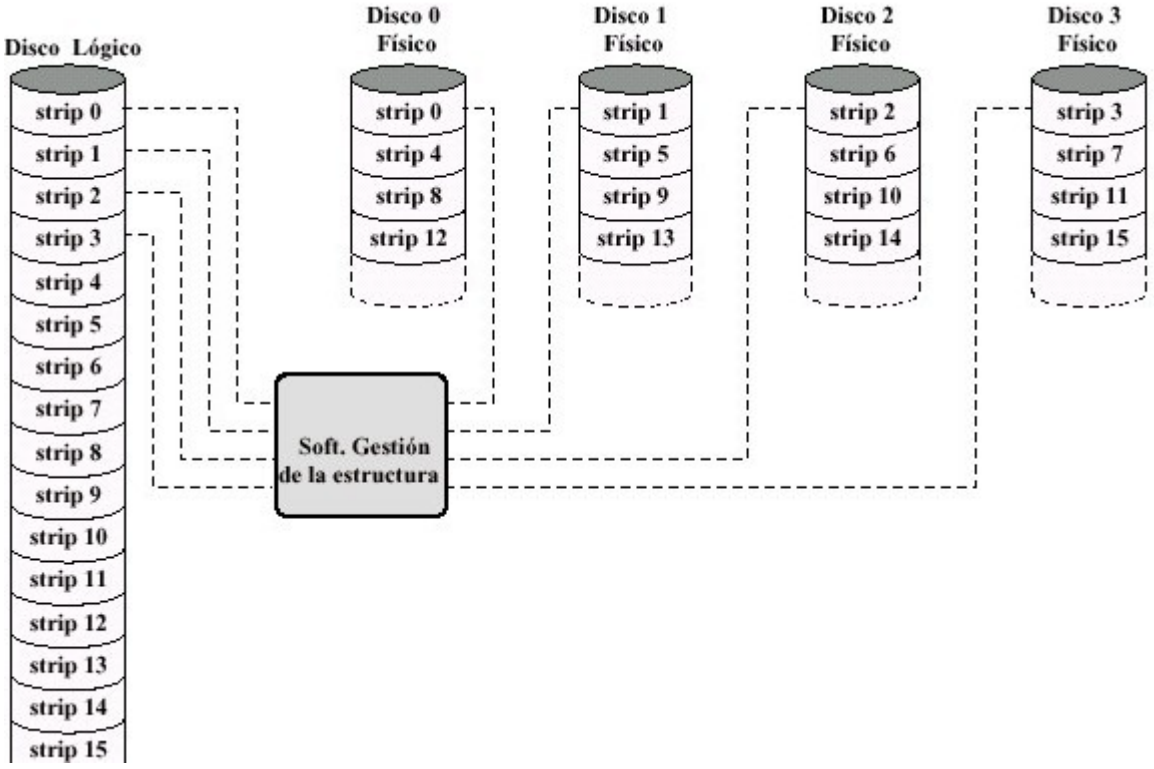
NIVEL 0 DE RAID

El nivel 0 de RAID no es un verdadero miembro de la familia RAID, porque no incluye redundancia para mejorar las prestaciones. Sin embargo, hay algunas aplicaciones, como algunas ejecuciones en supercomputadores, en las que las prestaciones y la capacidad son la preocupación primaria, y un costo bajo es más importante que mejorar la seguridad.



Para el RAID 0, los datos del usuario y del sistema están distribuidos a lo largo de todos los discos del conjunto. Esto tiene una notable ventaja frente al uso de un único y gran disco.

Si hay pendientes dos peticiones diferentes de E/S para dos bloques de datos diferentes, entonces es muy probable que los bloques pedidos estén en diferentes discos. Entonces, las dos peticiones se pueden emitir en paralelo, reduciendo el tiempo de cola de E/S.



Pero RAID 0, como todos los niveles RAID, va más lejos que una sencilla distribución de datos a través del conjunto de discos: los datos son *organizados en forma de tiras de datos a* través de los discos disponibles. Todos los datos del usuario y del sistema se ven como almacenados en un disco lógico. El disco se divide en tiras; estas tiras pueden ser bloques fisicos, sectores o alguna otra unidad. Las tiras se proyectan cíclicamente, en miembros consecutivos del conjunto. Un conjunto de tiras lógicamente consecutivas, que se proyectan exactamente sobre una misma tira en cada miembro del conjunto, se denomina «franja». En un conjunto de n discos, las primeras n tiras lógicas (una franja) se almacenan físicamente en la primera tira de cada uno de los n discos, las segundas n tiras lógicas, se distribuyen en la segunda tira de cada disco, etc. La ventaja de esta

disposición es que si una única petición de E/S implica a varias tiras lógicas contiguas, entonces las n tiras de esta petición se pueden gestionar en paralelo, reduciendo considerablemente el tiempo de transferencia de E/S.
En la figura anterior se indica como el software de gestión de un conjunto proyecta el espacio del disco físico sobre el disco 1ógico. Este software se puede ejecutar, tanto en el subsistema de disco como en un computador anfitrión.

RAID 0 para alta capacidad de transferencia de datos
Las prestaciones de cualquiera de los niveles RAID dependen críticamente de los patrones de petición del sistema anfitrión y de la distribución de los datos. Estas emisiones pueden ser más claramente direccionadas en RAID 0, donde el impacto de la redundancia no interfiere con el análisis. Primero, consideremos el uso de RAID 0 para lograr una velocidad de transferencia de datos alta. Se deben cumplir dos requisitos para que las aplicaciones tengan una velocidad de transferencia alta. Primero, debe existir una capacidad de transferencia alta en todo el camino entre la memoria del anfitrión y las unidades de disco individuales. Esto incluye controladores de buses internos, buses de E/S del anfitrión, adaptadores de E/S, y buses de memoria del anfitrión.
El segundo requisito es que la aplicación debe hacer peticiones de E/S que se distribuyan eficientemente sobre el conjunto de discos. Esta condición se satisface si la petición típica es de una gran cantidad de datos lógicamente contiguos, comparados con el tamaño de una cinta. En este caso, una única petición de E/S implica la transferencia paralela de datos desde varios discos, aumentando la velocidad efectiva de transferencia en comparación con la de un único disco.

RAID 0 para altas frecuencias de petición de EIS
En los entornos orientados a transacciones, el usuario se suele preocupar más del tiempo de respuesta que de la velocidad de transferencia. Para una petición individual de E/S de una pequeña cantidad de datos, el tiempo de E/S está dominado por el movimiento de las cabezas del disco (tiempo de búsqueda) y el movimiento del disco (latencia, rotacional),
En un entorno de transacción, puede haber cientos de peticiones de E/S por segundo. Un conjunto de discos puede proporcionar velocidades altas de ejecución de E/S, balanceando la carga de E/S a través de los distintos discos. El balanceo de la carga efectiva se consigue solamente si hay varias peticiones de E/S pendientes. Esto, por turnos, implica que hay varias aplicaciones independientes, o una única aplicación orientada a transacción que es capaz generar varias peticiones de E/S asíncronas. Las prestaciones también se verán influidas p el tamaño de la franja. Si la franja es relativamente grande, de forma que una única petición de E/S sólo implique un único acceso a disco, entonces las peticiones de E/S que están esperando pueden ser tratadas en paralelo, reduciendo el tiempo en cola para cada petición.

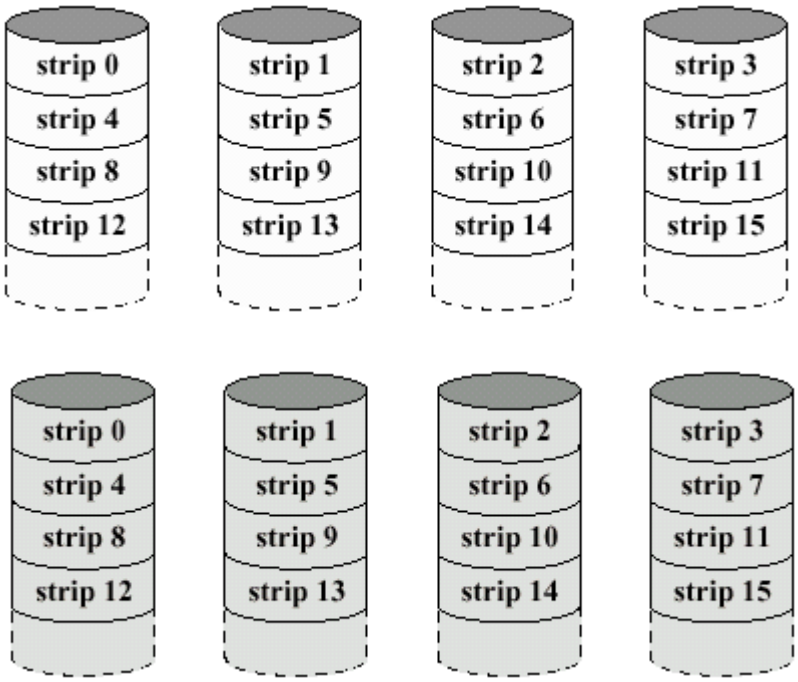
NIVEL 1 DE RAID

RAID 1 se diferencia de los niveles 2 a 5 en cómo se consigue la redundancia. En estos otros esquemas RAID, se usan algunas formas de cálculo de paridad para introducir redundancia en RAID 1, la redundancia se logra con el sencillo recurso de duplicar todos los datos. Según muestra la figura, se hace una distribución de datos, como en el RAID 0. Pero en e caso, cada franja lógica se proyecta en dos discos físicos separados, de forma que cada disco del conjunto tiene un disco espejo que contiene los mismos datos.
En la organización RAID 1 hay una serie de aspectos positivos:

1. Una petición de lectura puede ser servida por cualquiera de los discos que contienen los datos pedidos; cualquiera de ellos implica un tiempo de búsqueda mínimo más latencia rotacional.
2. Una petición de escritura requiere que las dos tiras correspondientes se actualicen y esto se puede hacer en paralelo. Entonces, el resultado de la escritura viene determinado por la

menos rápida de las dos escrituras (es decir, la que conlleva el mayor tiempo de búsqueda más la latencia rotacional). Sin embargo, en RAID 1 no hay <penalización en la escrituras>. Los niveles RAID del 2 al 5 implican el uso de bits paridad. Por tanto, cuando se actualiza una única tira, el software de gestión del conjunto debe calcular y actualizar primero los bits de paridad, así como actualizar la tira en cuestión.

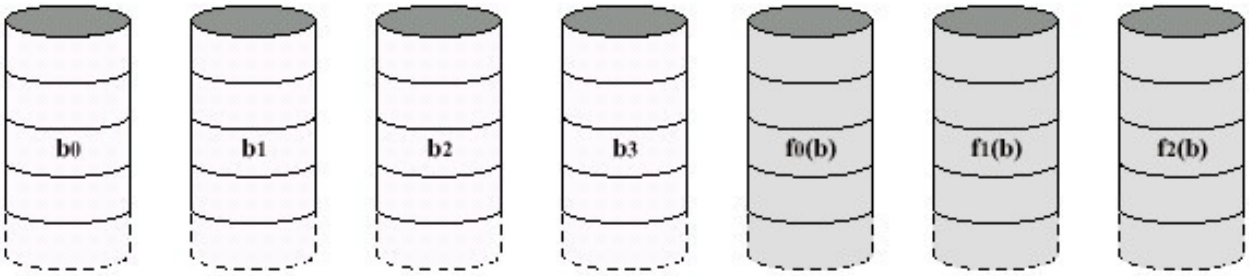
3. La recuperación tras un fallo es sencilla. Cuando una unidad falla, se puede acceder a los datos desde la segunda unidad.



La principal desventaja es el costo: requiere el doble del espacio de disco del disco lógico que puede soportar. Debido a esto, una configuración RAID 1 posiblemente está limitada a unidades que almacenan el software del sistema y los datos, y otros ficheros altamente críticos. En estos casos, RAID proporciona una copia de seguridad en tiempo real de todos los datos de forma que, en caso de fallo de disco, todos los datos críticos están inmediatamente disponibles.
En un entorno orientado a transacciones, RAID 1 puede conseguir altas velocidades petición de E/S si la mayor parte de las peticiones son lecturas. En esta situación, las prestaciones de RAID 1 son próximas al doble de las de RAID 0. Sin embargo, si una parte importante de las peticiones de E/S son peticiones de escritura, entonces la ganancia en prestaciones sobre RAID 0 puede no ser significativa. RAID 1 puede también proporcionar una mejora en las prestaciones de RAID 0 en aplicaciones de transferencia intensiva de datos con un alto porcentaje de lecturas. Se produce una mejora, si la aplicación puede dividir cada petición de lectura de forma que ambos miembros del disco participen.

NIVEL 2 DE RAID

Los niveles 2 y 3 de RAID usan una técnica de acceso paralelo. En un conjunto de acceso paralelo, todos los discos miembros participan en la ejecución de cada petición de E/S. Normalmente, el giro de cada unidad individual está sincronizado de forma que cada cabeza de disco está en la misma posición en cada disco en un instante dado.



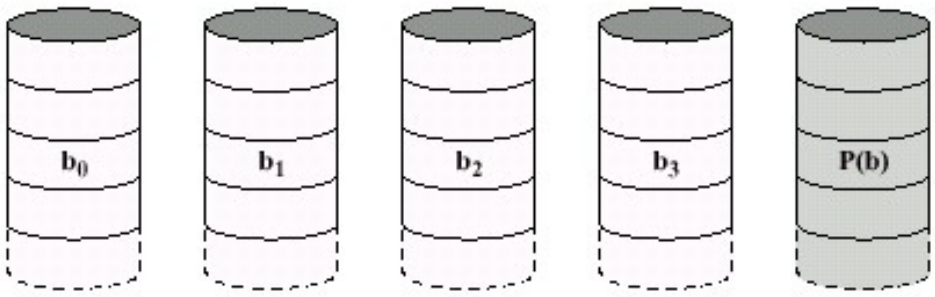
Como en los otros esquemas RAID, se usa la descomposición de datos en tiras. En el caso de RAID 2 y 3, las tiras son muy pequeñas a menudo, tan pequeñas como un único byte o palabra. Con RAID 2, el código de corrección de errores se calcula a partir de los bits de cada disco, y los bits del código se almacenan en las correspondientes posiciones de bit en varios discos de paridad. Normalmente, se usa el código Hamming, que permite corregir errores en un bit y detectar errores en dos bits.

Aunque RAID 2 requiere menos discos que RAID 1, es todavía bastante caro. El número de discos redundantes es proporcional al logaritmo del número de discos de datos. En una sola lectura se accede a todos los discos simultáneamente. El controlador del conjunto proporciona los datos pedidos y el código de corrección de errores asociado. Si hay un error en un solo bit, el controlador lo puede reconocer y corregir instantáneamente, con lo que el tiempo de acceso a lectura no se ralentiza. En una escritura sencilla, la operación de escritura debe acceder a todos los discos de datos y de paridad.

RAID 2 debería ser solamente una elección efectiva en un entorno en el que haya muchos errores de disco. Si hay una alta seguridad en los discos individuales y en las unidades de disco, RAID 2 es excesivo y no se implementa.

NIVEL 3 DE RAID

RAID 3 se organiza de manera similar a RAID 2. La diferencia es que RAID 3 requiere sólo un disco redundante, sin importar lo grande que sea el conjunto de discos. RAID 3 utiliza un acceso.paralelo, con datos distribuidos en pequeñas tiras. En vez de un código de corrección de errores, se calcula un sencillo bit de paridad para el conjunto de bits individuales que están en la misma posición en todos los discos de datos.



Redundancia

En el caso de un fallo en una unidad, se accede a la unidad de paridad y se reconstruyen los datos desde el resto de los dispositivos. Una vez se sustituye la unidad que ha fallado, los datos que faltan se restauran en la nueva'unidad y se reanuda la operación.

La reconstrucción de los datos es bastante sencilla. Consideremos un conjunto de cinco discos, de los que de X0 a X3 contienen datos, y X4 es el disco de paridad. La paridad para el i-ésimo bit se calcula de la siguiente forma:

$$X4(i) = X3(i) \oplus X2(i) \oplus X1(i) \oplus X0(i)$$

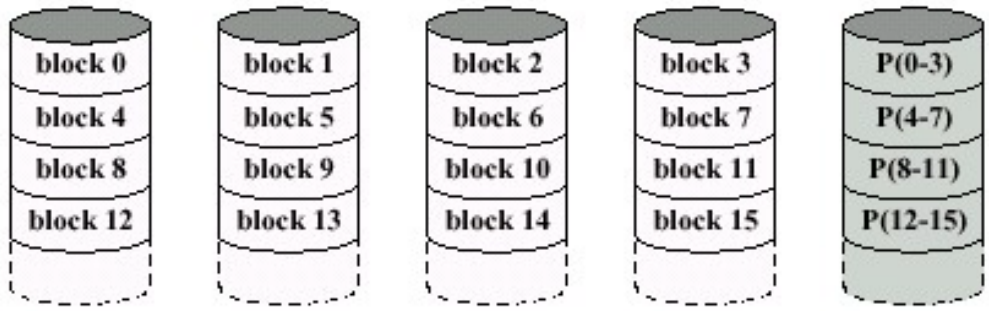
Supongamos que la unidad X1 ha fallado. Si sumamos X4(i) ⊕ X1(i) a ambos miembros de la ecuación, tenemos que:

$$X1(i) = X4(i) \oplus X3(i) \oplus X2(i) \oplus X0(i)$$

Por lo tanto, se puede regenerar el contenido de cualquier tira de datos en X1 a partir del contenido de las correspondientes tiras del resto de los discos del conjunto. Este principio es válido para los niveles 3 a 6 de RAID.
Caso de que un disco falle, todos los datos estarán todavía disponibles en lo que se de- nornina modo reducido. En este modo, para lecturas, los datos que faltan se recuperan «al vuelo» con la operación exclusive-or. Cuando se escriben datos en un conjunto RAID 3 reducido, se debe mantener la consistencia de la paridad para regeneraciones posteriores.
Volviendo al funcionamiento global, se requiere que el disco que ha fallado se reemplace y se regenere todo su contenido en el nuevo disco.

Prestaciones
Puesto que los datos se dividen en tiras muy pequenas, RAID 3 puede conseguir velocidades de transferencia de datos muy altas. Cualquier petición de E/S implicará una transferencia de datos paralela desde todos los discos de datos. Para grandes transferencias, la mejora de prestaciones es especialmente notable. Por otra parte, sólo se puede ejecutar a la vez una petición de E/S. Por tanto, en un entorno orientado a transacciones, el rendimiento sufre.

NIVEL 4 DE RAID
Los niveles 4 y 5 de RAID usan una técnica de acceso independiente. En un conjunto de acceso independiente, cada disco opera independientemente, de forma que peticiones de E/S separadas se atienden en paralelo. Debido a esto, son más adecuados los conjuntos de acceso independiente para aplicaciones que requieren velocidades de petición de E/S altas, y son menos adecuados para aplicaciones que requieren velocidades altas de transferencia de datos.
Como en otros esquemas RAID, se usan tiras de datos. En el caso de RAID 4 y 5, las tiras son relativamente grandes. Con RAID 4 se calcula una tira de paridad, bit a bit, a partir de las correspondientes tiras de cada disco de datos, y los bits de paridad se almacenan en la correspondiente tira del disco de paridad.



RAID 4 lleva consigo una penalización en la escritura cuando se realiza una petición de escritura de E/S pequeña. Cada vez que se realiza una escritura, el software de gestión del conjunto debe actualizar, no sólo los datos del usuario, sino también los bits de paridad correspondientes. Consideremos un conjunto de cinco unidades en las que de X0 a X3 contienen datos y X4 es el disco de paridad. Supongamos que se realiza una escritura que implica sólo,una tira del disco X1. Inicialmente, para cada bit i, tenemos la siguiente relación:

$$X4(i) = X3(i) \oplus X2(i) \oplus X1(i) \oplus X0(i)$$

Después de la actualización, indicamos con prima los bits que han sido alterados:

$X4'(i) = X3(i) \oplus X2(i) \oplus X1'(i) \oplus X0(i)$

$= X3(i) \oplus X2(i) \oplus X1'(i) \oplus X0(i) \oplus X1(i) \oplus X1(i)$

$= X4(i) \oplus X1(i) \oplus X1'(i)$

Para calcular la nueva paridad, el software de gestión del conjunto debe leer la antigua tira del usuario y la antigua tira de paridad. Entonces, se pueden actualizar estas dos tiras con nuevos datos y calcular la nueva paridad. Por tanto, cada escritura de una tira implica dos lecturas y dos escrituras.

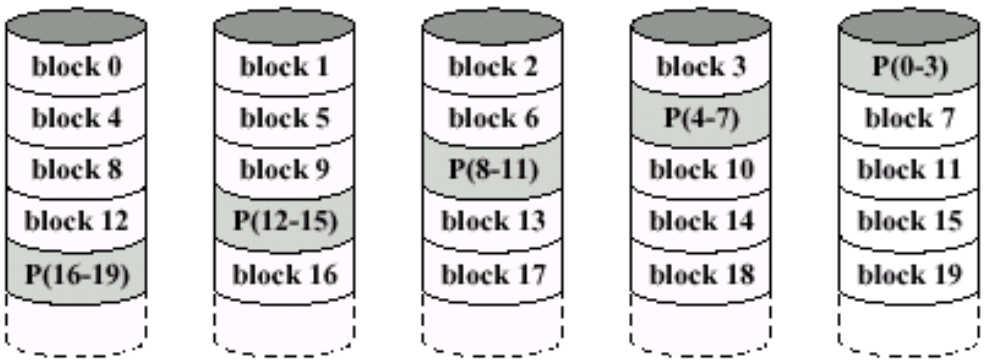
En el caso de una escritura de E/S de mayor tamaño, que implique tiras en todas las unidades de disco, la paridad se puede obtener fácilmente con un cálculo, usando solamente los nuevos bits de datos. Por tanto, la unidad de paridad puede ser actualizada en paralelo con las unidades de datos, y no habrá lecturas o escrituras extra.

En cualquier caso, cada operación de escritura implica al disco de paridad que, por consiguiente, se convertirá en un cuello de botella.

NIVEL 5 DE RAID

RAID 5 está organizado de manera similar a RAID 4. La diferencia es que RAID 5 distribuye las tiras de paridad a lo largo de todos los discos. Una distribución típica es un esquema cíclico, como se muestra en la figura. Para un conjunto de n discos, la tira de paridad está en diferentes discos para las primeras n tiras, y este patrón se repite.

La distribución de las tiras de paridad a lo largo de todas las unidades, evita el potencial cuello de botella de E/S encontrado en RAID 4.

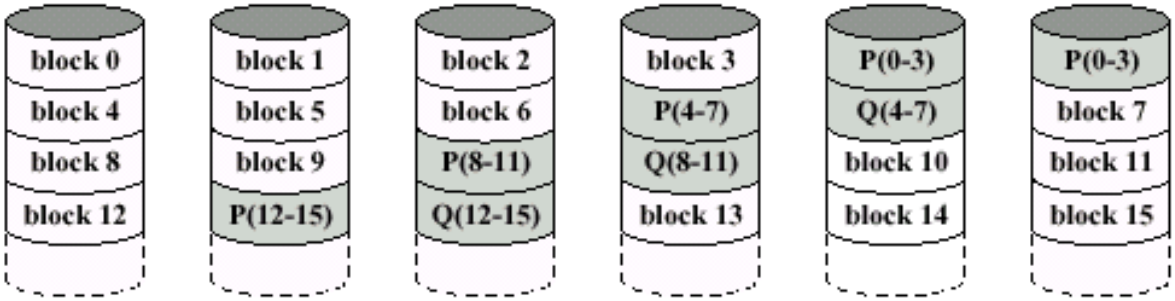


NIVEL 6 DE RAID

El nivel 6 de RAID se introdujo en un artículo de los investigadores de Berkeley [KATZ89]. En el esquema de] nivel 6 de RAID, se hacen dos cálculos de paridad distintos, que se almacenan en bloques separados en distintos discos. Por tanto, un conjunto RAID 6 cuyos datos requieran N discos consta de N + 2 discos.

En la figura se ilustra este esquema. P y Q son dos algoritmos de comprobación de datos distintos. Uno de los dos calcula la exclusive-OR usada en los niveles de 4 y 5 de RAID, pero el otro es un algoritmo de comprobación de datos independiente. Esto hace posible la regeneración de los datos, incluso si dos de los discos que contienen los datos de los usuarios fallan.

La ventaja del RAID 6 es que proporciona una disponibilidad de los datos extremadamente alta. Tendrían que fallar tres discos en el intervalo MTTR (tiempo medio de reparación) para no poder disponer de los datos. Por otra parte, RAID 6 incurre en una penalización de escritura, ya que cada escritura afecta a dos bloques de paridad.


block 0
block 4
block 8
block 12
block 1
block 5
block 9
P(12-15)
block 2
block 6
P(8-11)
Q(12-15)
block 3
P(4-7)
Q(8-11)
block 13
P(0-3)
Q(4-7)
block 10
block 14
P(0-3)
block 7
block 11
block 15