

RAID

Gerardo

RAID

- Redudant Array of Independent Disk, Arreglo Redudante de Discos Independientes.
- Método por el cual discos independientes, desde la perspectiva de los usuarios y aplicaciones, aparecen como uno solo.

Ventajas

- Mejora el rendimiento.
- Mayor capacidad de almacenamiento.
- Se puede transparentar el almacenamiento de datos en caso de fallo en un disco.

RAID Software/Hardware

- Se puede tener RAID por software, utilizando el CPU para las operaciones del arreglo.
- Hardware, utiliza procesadores especializados en el controlador de los discos para administrarlos.

Historia

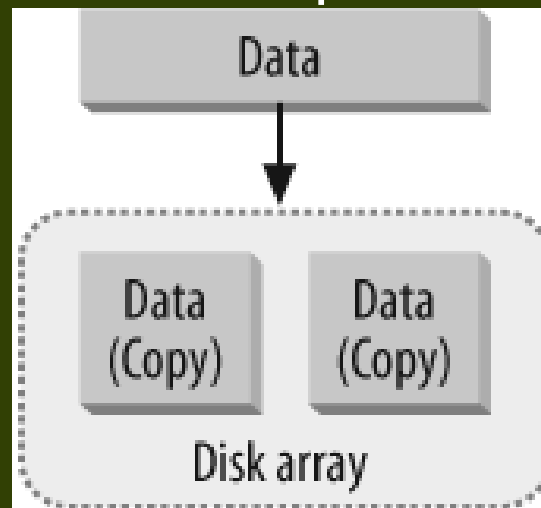
- El término RAID fue acuñado en Berkeley en 1988 por David A. Patterson, Garth A. Gibson y Randy H. Katz en el artículo “A Case for Redudant Arrays of Inexpensive Disk (RAID)”.

Redundancia

- Es una característica que permite a un arreglo seguir funcionando a pesar de una falla en un disco.
- No todos los niveles de RAID soportan esta característica

Mirroring (Duplicación)

- Aparecen dos formas básicas de redundancia a lo largo de la especificación RAID.
- El primero se logra con un proceso llamado duplicación (*mirroring*) de disco.
- *Mirroring* duplica los datos en cada disco del arreglo.
- Cada disco miembro contiene los mismos datos y tiene un papel igual en el arreglo.
- En caso de falla del disco, los datos se pueden leer desde los discos restantes

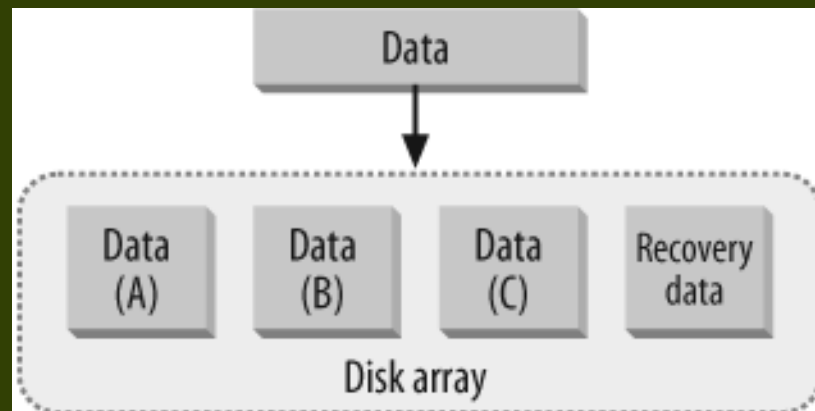


Mirroring

- Se lleva a cabo una mejora en el rendimiento de la lectura producto de la duplicación.
- Cuando el arreglo está funcionando normalmente, lo que significa que ningún disco ha fallado, los datos se pueden leer en paralelo desde cada disco en el espejo.
- El resultado es que las lecturas pueden producir un rendimiento lineal basado en la cantidad de discos en la matriz.
- Un espejo de dos discos podría producir velocidades de lectura de hasta dos veces más que un solo disco.
- En la práctica no verá un aumento en el rendimiento de lectura tan dramático. Eso se debe a que muchos otros factores, incluido el rendimiento del sistema de archivos y la distribución de datos, también afectan el rendimiento.
- Desafortunadamente, la duplicación también significa que los datos deben escribirse dos veces, una vez en cada disco del arreglo.
- El resultado es un rendimiento de escritura ligeramente más lento, en comparación con el de un solo disco o arreglo que no se encuentra en *mirroring*.

Paridad

- Los algoritmos de paridad son otro método para lograr redundancia.
- Cuando los datos se escriben en el arreglo la información de recuperación se escribe en un disco separado.
- Si falla un disco, los datos originales se pueden reconstruir a partir de la información de paridad y los datos restantes.



Degraded

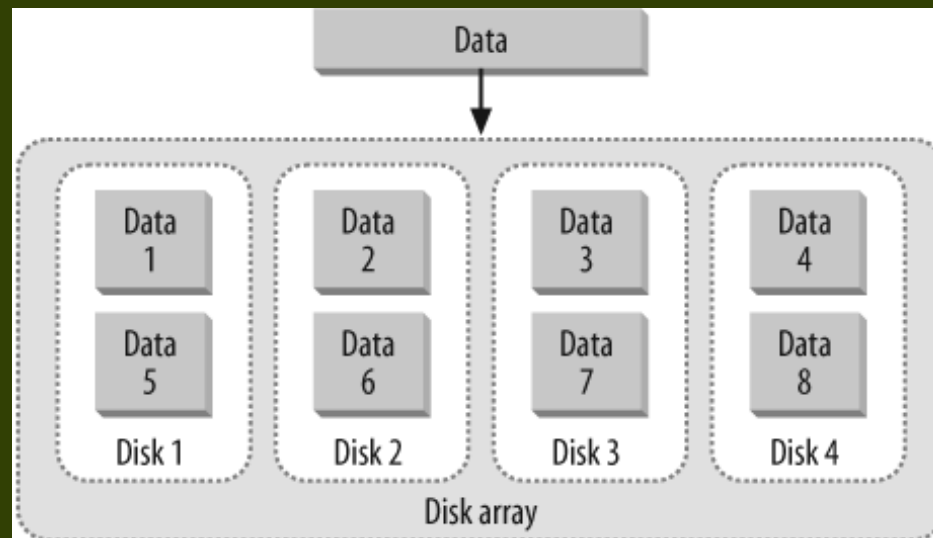
- Degraded describe un arreglo que admite redundancia, pero tiene uno o más discos con fallos.
- El arreglo sigue siendo operativo, pero su fiabilidad y, en algunos casos, su rendimiento, se ve disminuido.
- Cuando un arreglo está en modo degradado, una falla adicional en el disco generalmente indica pérdida de datos, aunque ciertos tipos de arreglos pueden soportar múltiples fallas de disco.

Reconstrucción, resincronización y recuperación.

- Cuando se reemplaza un disco que falló, inicia un proceso de recuperación.
- Los términos reconstrucción, resincronización, y recuperación se usan de manera indistinta para describir el proceso de recuperación.
- Durante la recuperación, los datos se copian textualmente en el nuevo disco (si se usó la duplicación) o se reconstruyen usando la información de paridad proporcionada por los discos restantes (si se utilizó paridad).
- El proceso de recuperación generalmente pone una carga adicional en los recursos del sistema. La recuperación puede ser automatizada tanto por hardware como por software, siempre que haya suficiente hardware (discos) disponible para reparar un arreglo sin la intervención del usuario.
- Cada vez que se crea un arreglo redundante, se realiza un proceso de recuperación inicial. Este proceso asegura que todos los discos estén sincronizados. Es parte de las operaciones normales de RAID y no indica ningún error de hardware o software.

Striping

- *Striping* es un método mediante el cual los datos se distribuyen en múltiples discos.
- Se escribe una cantidad fija de datos en cada disco. El primer disco del arreglo no se reutiliza hasta que se escribe una cantidad igual de datos en cada uno de los otros discos.
- Esto se traduce en un mejor rendimiento de lectura y escritura, ya que los datos se graban en más de una unidad a la vez.
- Algunos arreglos que almacenan datos en bandas también admiten redundancia a través de la paridad de disco.
- RAID-0 define un arreglo seccionado sin redundancia, lo que resulta en un rendimiento de lectura y escritura extremadamente rápido, pero no tiene un método para sobrevivir a una falla de disco. No todos los tipos de arreglos admiten *striping*.

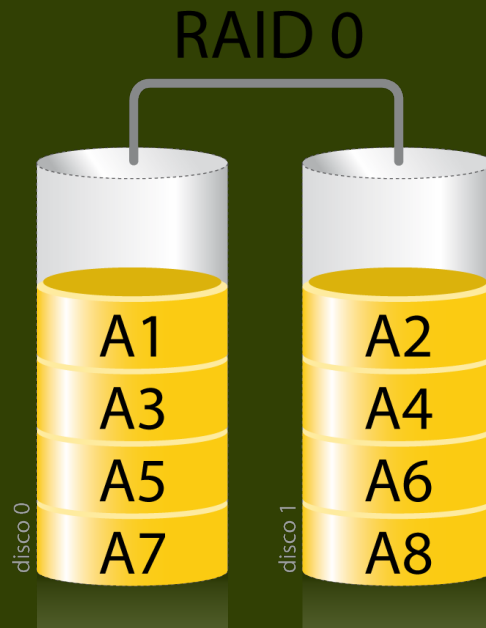


RAID-0: Striping

- Un RAID 0 (también llamado conjunto dividido, volumen dividido, volumen seccionado) distribuye los datos equitativamente entre dos o más discos (usualmente se ocupa el mismo espacio en dos o más discos) sin información de paridad que proporcione redundancia.
- Es importante señalar que el RAID 0 no era uno de los niveles RAID originales y que no es redundante.
- El RAID 0 se usa habitualmente para proporcionar un alto rendimiento de escritura ya que los datos se escriben en dos o más discos de forma paralela, aunque un mismo archivo solo está presente una vez en el conjunto.
- RAID 0 también puede utilizarse como forma de crear un pequeño número de grandes discos virtuales a partir de un gran número de pequeños discos físicos.
- Un RAID 0 puede ser creado con discos de diferentes tamaños, pero el espacio de almacenamiento añadido al conjunto estará limitado por el tamaño del disco más pequeño (por ejemplo, si un disco de 450 GB se divide con uno de 100 GB, el tamaño del conjunto resultante será sólo de 200 GB, ya que cada disco aporta 100 GB).
- Una buena implementación de un RAID 0 dividirá las operaciones de lectura y escritura en bloques de igual tamaño, por lo que distribuirá la información equitativamente entre los dos discos.

RAID-0

- También es posible crear un RAID 0 con más de dos discos, si bien, la fiabilidad del conjunto será igual a la fiabilidad media de cada disco entre el número de discos del conjunto; es decir, la fiabilidad total —medida como MTTF o MTBF— es (aproximadamente) inversamente proporcional al número de discos del conjunto (pues para que el conjunto falle es suficiente con que lo haga cualquiera de sus discos).



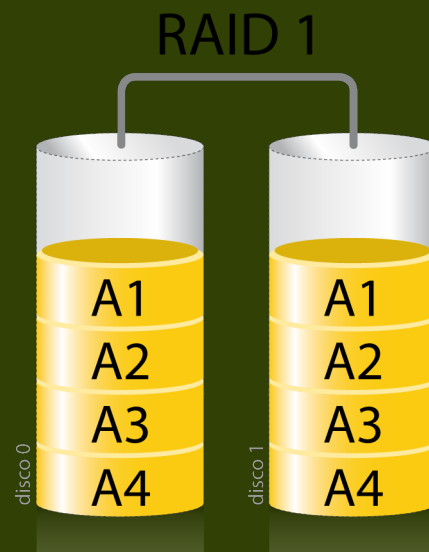
De JaviMZN - Trabajo propio, CC BY 3.0,
<https://commons.wikimedia.org/w/index.php?curid=9062448>

RAID-1: Mirroring

- Un RAID 1 crea una copia exacta (o espejo) de un conjunto de datos en dos o más discos. Esto resulta útil cuando queremos tener más seguridad desaprovechando capacidad, ya que si perdemos un disco, tenemos el otro con la misma información.
- Un conjunto RAID 1 sólo puede ser tan grande como el más pequeño de sus discos. Un RAID 1 clásico consiste en dos discos en espejo, lo que incrementa exponencialmente la fiabilidad respecto a un solo disco; es decir, la probabilidad de fallo del conjunto es igual al producto de las probabilidades de fallo de cada uno de los discos (pues para que el conjunto falle es necesario que lo hagan todos sus discos).
- Además, dado que todos los datos están en dos o más discos, con hardware habitualmente independiente, el rendimiento de lectura se incrementa aproximadamente como múltiplo lineal del número de copias; es decir, un RAID 1 puede estar leyendo simultáneamente dos datos diferentes en dos discos diferentes, por lo que su rendimiento se duplica. Para maximizar los beneficios sobre el rendimiento del RAID 1 se recomienda el uso de controladoras de disco independientes, una para cada disco (práctica que algunos denominan splitting o duplexing).
- Como en el RAID 0, el tiempo medio de lectura se reduce, ya que los sectores a buscar pueden dividirse entre los discos, bajando el tiempo de búsqueda y subiendo la tasa de transferencia, con el único límite de la velocidad soportada por la controladora RAID. Sin embargo, muchas tarjetas RAID 1 IDE antiguas leen sólo de un disco de la pareja, por lo que su rendimiento es igual al de un único disco. Algunas implementaciones RAID 1 antiguas también leen de ambos discos simultáneamente y comparan los datos para detectar errores.

RAID-1

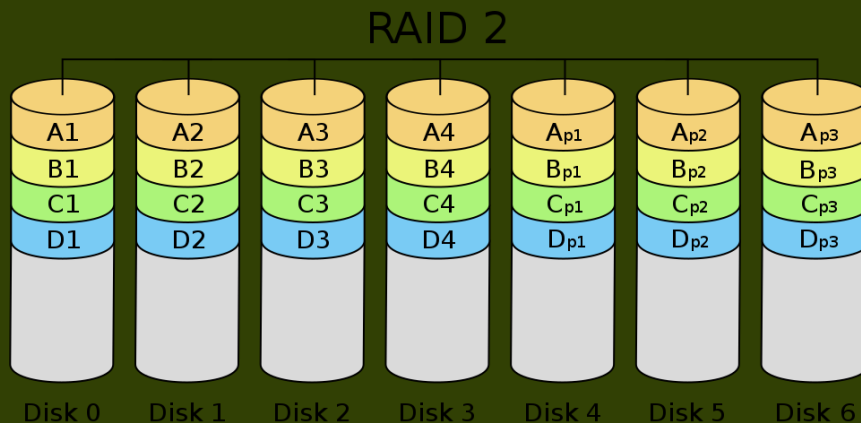
- Al escribir, el conjunto se comporta como un único disco, dado que los datos deben ser escritos en todos los discos del RAID 1. Por tanto, el rendimiento de escritura no mejora.
- El RAID 1 tiene muchas ventajas de administración. Por ejemplo, en algunos entornos 24/7, es posible “dividir el espejo”: marcar un disco como inactivo, hacer una copia de seguridad de dicho disco y luego “reconstruir” el espejo. Esto requiere que la aplicación de gestión del conjunto soporte la recuperación de los datos del disco en el momento de la división.



De JaviMZN - Trabajo propio, CC BY 3.0,
<https://commons.wikimedia.org/w/index.php?curid=9062568>

RAID-2

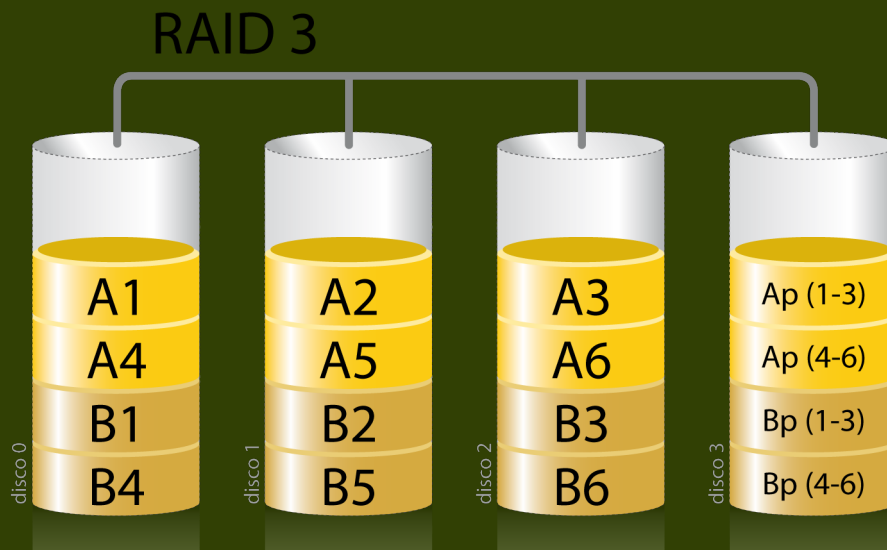
- Distribuye los datos entrelazados a nivel de bit. El código de error se intercala a través de varios discos también a nivel de bit, el código de error se calcula con el código de Hamming.
- Todo giro del cabezal de disco se sincroniza y los datos se distribuyen en bandas de modo que cada bit secuencial está en una unidad diferente.
- La paridad de Hamming se calcula a través de los bits correspondientes y se almacena en al menos un disco de paridad.
- Este nivel es sólo significativo a nivel histórico y teórico, ya que actualmente no se utiliza.



De User:knakts - File:RAID_6.svg, CC BY-SA 3.0,
<https://commons.wikimedia.org/w/index.php?curid=10648378>

RAID-3

- Un RAID 3 divide los datos a nivel de bytes en lugar de a nivel de bloques . Los discos son sincronizados por la controladora para funcionar al mismo tiempo.
- Permite tasas de transferencias extremadamente altas.
- Un RAID 3 necesitaría un mínimo de tres discos, utilizando uno para datos de paridad.
- En el ejemplo de la imagen, una petición del bloque «A» formado por los bytes A1 a A6 requeriría que los tres discos de datos buscaran el comienzo (A1) y devolvieran su contenido. Una petición simultánea del bloque «B» tendría que esperar a que la anterior concluyese.



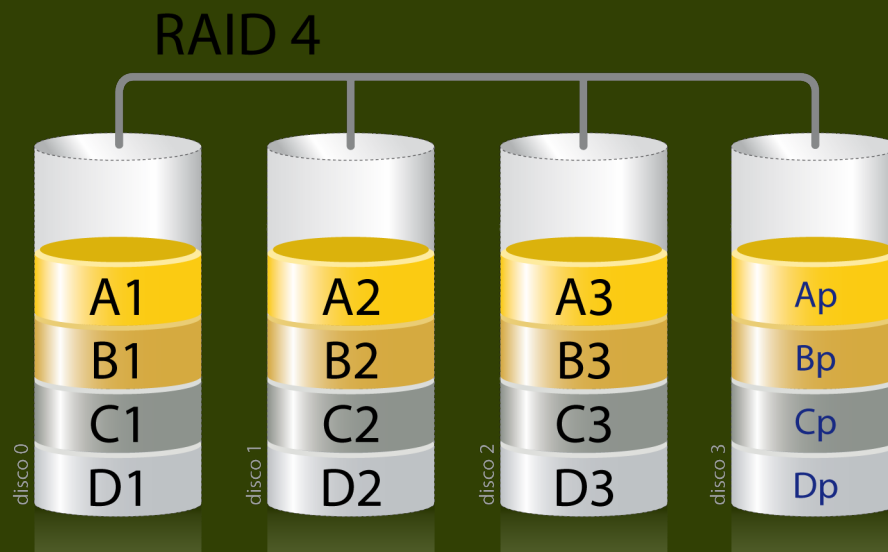
De JaviMZN - Trabajo propio, CC BY 3.0,
<https://commons.wikimedia.org/w/index.php?curid=9062574>

RAID-4: Dedicated Parity

- Un RAID 4, también conocido como IDA (acceso independiente con discos dedicados a la paridad), usa división a nivel de bloques con un disco de paridad dedicado.
- Necesita un mínimo de 3 discos físicos.
- El RAID 4 es parecido al RAID 3 excepto porque divide a nivel de bloques en lugar de a nivel de bytes.
- Esto permite que cada miembro del conjunto funcione independientemente cuando se solicita un único bloque.
- Si la controladora de disco lo permite, un conjunto RAID 4 puede servir varias peticiones de lectura simultáneamente.
- En principio también sería posible servir varias peticiones de escritura simultáneamente, pero al estar toda la información de paridad en un solo disco, éste se convertiría en el cuello de botella del conjunto.

RAID-4

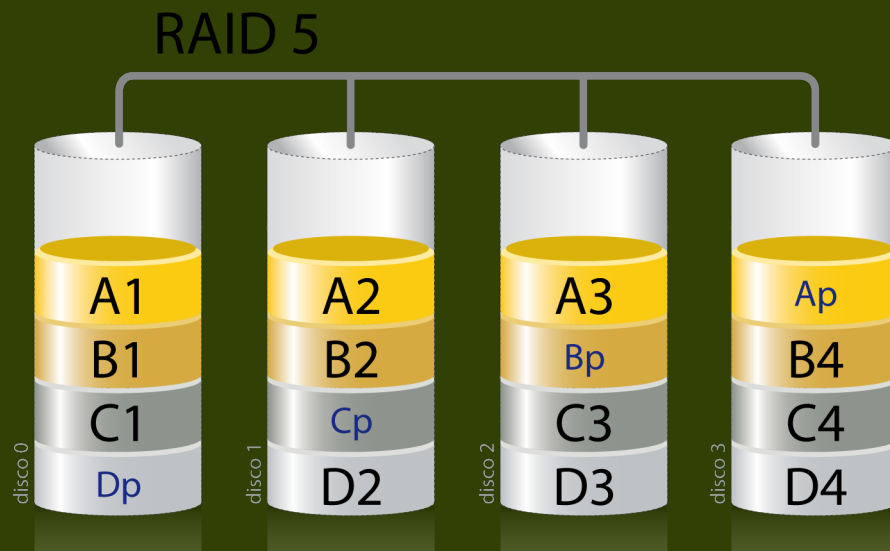
- Una petición del bloque «A1» sería servida por el disco 0. Una petición simultánea del bloque «B1» tendría que esperar, pero una petición de «B2» podría atenderse concurrentemente.



De JaviMZN - Trabajo propio, CC BY 3.0, <https://commons.wikimedia.org/w/index.php?curid=9062587>

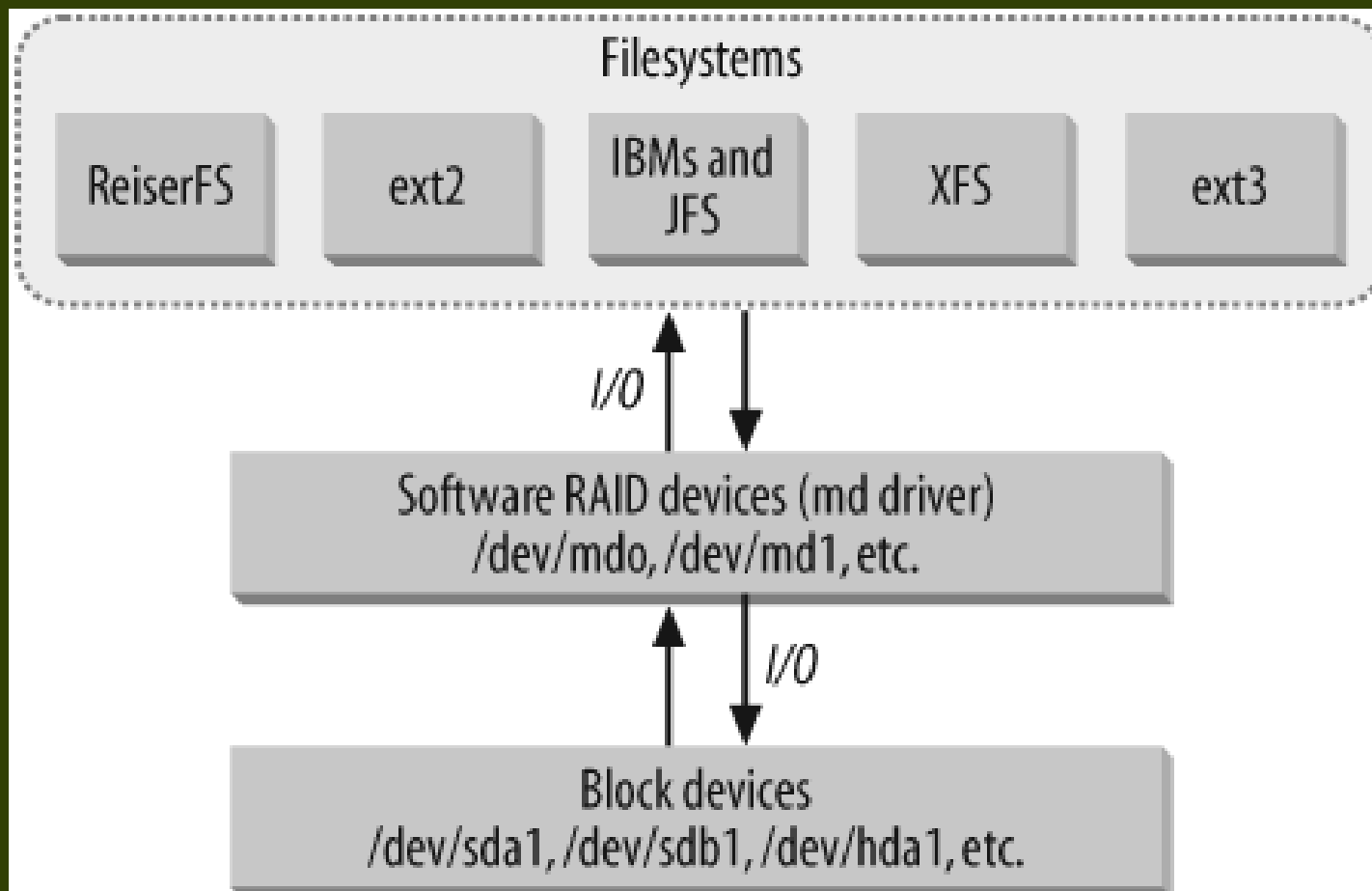
RAID-5: Distributed Parity

- Es similar a RAID-4 solo que la información de la paridad se divide a través de todos los discos en el arreglo.
- Esto reduce los cuellos de botella inherentes en la escritura de la paridad.



De JaviMZN - Trabajo propio, CC BY 3.0, <https://commons.wikimedia.org/w/index.php?curid=9062587>

RAID en Linux



Actividad

- Elegir un tipo de RAID e implementarlo en Linux.

Raid 1 en Linux

- Instalar mdadm
 - `sudo apt-get install mdadm`
- Desmontar particiones
 - `umount <particiones>`
- Agregar la bandera fd (linux raid auto) a las particiones (con fdisk)
 - `fdisk /dev/dispositivo`
- Crear arreglo
 - `sudo mdadm --create /dev/md0 --level=1 --raid-devices=3 /dev/sdc1 /dev/sdd1 /dev/sde1`
- Formatear el RAID
 - `mkfs.ext4 /dev/md0`
- Montar el RAID
 - `mount /dev/md0 /media/<arreglo>`

Remover un arreglo

- Detener el RAID
- `umount /dev/md0`
- `mdadm --stop /dev/md0`
- `mdadm --zero-superblock /dev/sd[c-d]1`