

Sistemas RAID

Conceptos básicos

Programa

- ¿Que es RAID?
- Particularidades
- *hardware vs. software*
- Niveles de RAID
- Comparando niveles
- Tolerancia a fallas
- Confiabilidad y disponibilidad
- Implementando en Linux (algunos ejemplos)

¿Que es RAID?

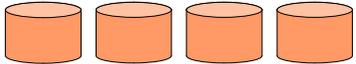
En términos generales y partiendo de su acrónimo en ingles **Redundant Array of Independent Disks (RAID)** es un sistema que permite combinar el almacenamiento de un grupo de dispositivos independientes, en una única unidad virtual de almacenamiento o múltiples unidades virtuales.

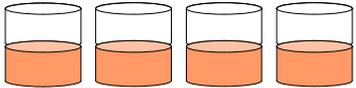
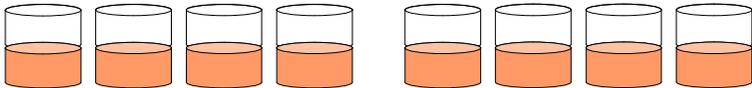
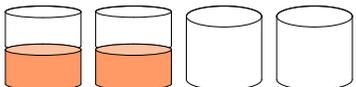
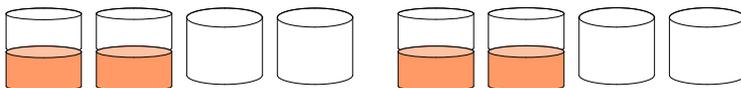


Por ejemplo se pueden combinar grupos de discos duros, grupos de dispositivos de estado solido (*solid-state drive*, SDD) o grupos de ambos!!!

Particularidades

- Mayor rendimiento y confiabilidad mediante lectura/escritura simultanea de datos en múltiples dispositivos (físicos o virtuales). Recordando que:

- Dispositivos físicos → ej: Grupo de discos duros 
- Dispositivos virtuales → ej: Particiones en un grupo de discos que podrían ser:

- En un grupo completo 
- En mas de un grupo 
- En parte de un grupo 
- En parte de mas de un grupo 

Particularidades

- Diferentes esquemas de lectura/escritura conocidos como niveles (*RAID levels*)
- El nivel a elegir depende de necesidades en cuanto a:
 - Rendimiento y redundancia
 - Costos de *hardware*
 - Capacidad almacenamiento (escalabilidad)
- Soluciones de RAID pueden estar basadas en *hardware* especializado o herramientas de *software*.

Hardware vs. Software

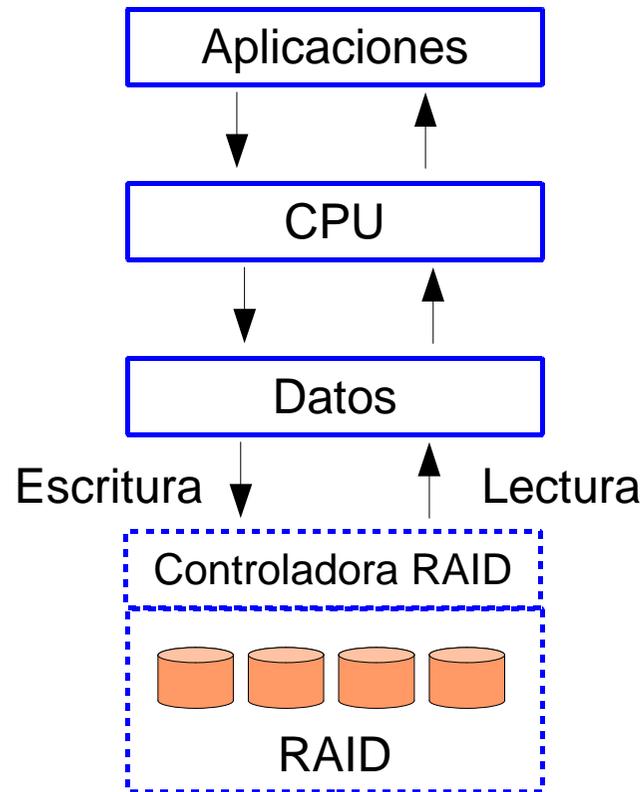
- *¿hardware o software?* Una de las primeras decisiones que se debe tomar.
- *hardware RAID*
 - Uso de procesadores dedicados (generalmente ubicados en controladoras de discos) para realizar las operaciones del arreglo.
- *software RAID*
 - Uso del CPU del computador para realizar operaciones del arreglo implementadas a nivel de *kernel*.

Hardware vs. Software

hardware RAID

- El arreglo es administrado por una controladora de disco especializado que contiene un *software* embebido (*firmware*) para RAID .
- La capa de *software* del computador accede a un único dispositivo virtual de almacenamiento. El arreglo esta oculto y es administrado por el controlador de RAID.
- Las soluciones de *hardware* RAID pueden ser:
 - Tarjetas controladoras RAID
 - Gabinetes externos conectados a puertos SCSI, *Fiber Channel*, otros.
 - Gabinetes externos conectados en red (*storage area network - SAN*)

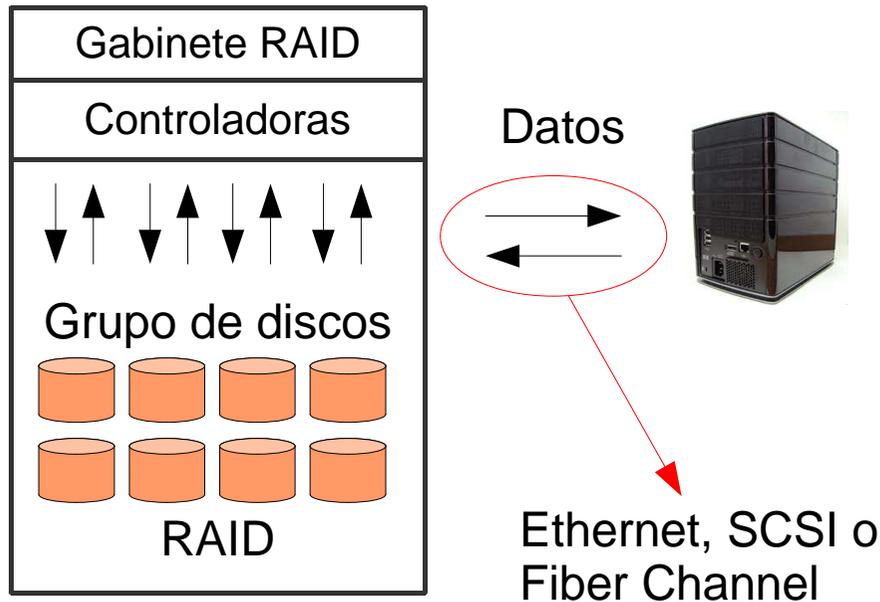
Hardware vs. Software



Tarjetas controladoras

- Directamente instaladas en el computador (ej. PCI), reciben la conexión de los dispositivos generalmente a través de interfaces estándar (ej. IDE, SATA, otros).
- Contiene un BIOS para administración, configuración y mantenimiento del RAID.
- Es importante asegurar que es soportada por el OS (ej. Linux)

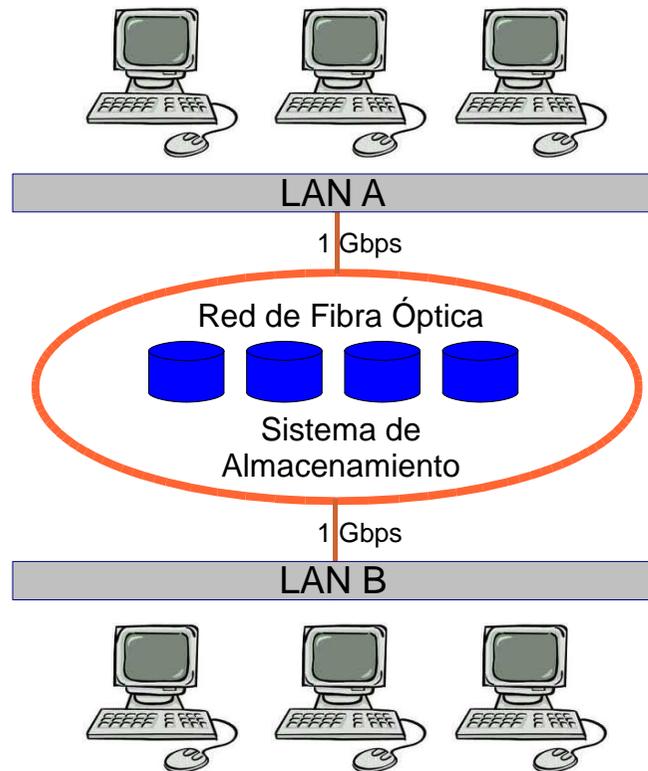
Hardware vs. Software



Gabinetes externos

- Conectados a través de puertos de alto rendimiento (ej. SCSI, *Fiber Channel*, otros)
- Aparecen como un punto de montaje externo (no requieren módulos especiales a nivel de *kernel*)
- Alto costo inicial y de mantenimiento (generalmente soluciones propietarias)

Hardware vs. Software



Storage Area Networks (SAN)

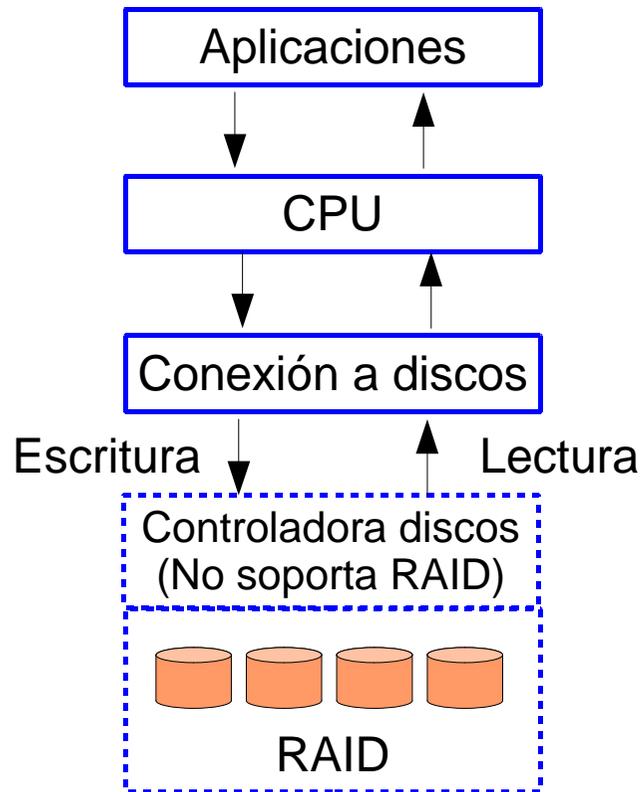
- Varias plataformas de almacenamiento interconectadas a través de una red de alta velocidad.
- Conectada a diferentes partes de la LAN aumentando el rendimiento hacia el sistema de almacenamiento.
- RAID es una parte vital de la SAN.
- Se esta convirtiendo en la tendencia en organizaciones distribuidas de gran escala.

Hardware vs. Software

software RAID

- El arreglo es administrado a nivel de *kernel*.
- El *kernel* mantiene la organización de los datos en varios discos mientras presenta un solo dispositivo virtual a la capa de aplicaciones.
- Se ha popularizado en las ultimas década debido a:
 - Mayor capacidad de CPU a bajos costos
 - Muchos sistemas operativos (ej. Linux) proveen soporte y funcionalidades de RAID como parte del *software*.

Hardware vs. Software



software RAID en Linux

- Muchas distribuciones soportan RAID de forma nativa a nivel de *kernel*.
- Herramientas maduras (ej. mdadm) que permiten crear, consultar, sincronizar y dar mantenimiento completo a los dispositivos RAID.

Niveles de RAID

- Diferentes aplicaciones requieren implementaciones de diferentes estructuras de RAID. Estas diferentes estructuras se conocen como niveles.
- Diferentes niveles de RAID ofrecen diversidad de compromiso entre rendimiento y redundancia.
- La selección del nivel adecuado requiere un alto entendimiento de las necesidades de sus aplicaciones y usuarios (incluyendo escalabilidad).



Por ejemplo es posible que según sus necesidades deba sacrificar rendimiento para lograr un RAID de mayor redundancia!!!

Niveles de RAID

Revisando conceptos

striping

- Técnica de segmentación lógica de los datos que luego son accedidos de forma secuencial en diferentes dispositivos de almacenamiento.
- Provee un mayor rendimiento en la velocidad de acceso a los datos almacenados en múltiples dispositivos.
- La falla de un dispositivo causa la pérdida de todos los datos.
- Según lo anterior el porcentaje de falla es la suma del porcentaje de falla de cada dispositivos.

Niveles de RAID

Revisando conceptos

mirroring

- Técnica en la cual se crean replicas en tiempo real del volumen lógico de un dispositivo en otros dispositivos físicos
- Ofrece alta disponibilidad de los datos debido a las múltiples replicas de los datos (redundancia).
- De forma adicional provee mejoras en el rendimiento de acceso a los datos (lectura) en múltiples dispositivos.
- Implementaciones de esta técnica representan un alto costo debido a la replica de cada dispositivo.

Niveles de RAID

Revisando conceptos

parity

- Se genera un conjunto de datos de redundancia a partir de dos o más conjuntos de datos primarios aplicando la función Booleana XOR ([ver apéndice A](#))
- Con los datos de redundancia se puede reconstruir los datos de algunos de los conjuntos de datos primarios.
- Aunque no implica duplicar por completo los datos primarios, esta técnica puede ocasionar bajo rendimiento en la velocidad de escritura en un RAID.

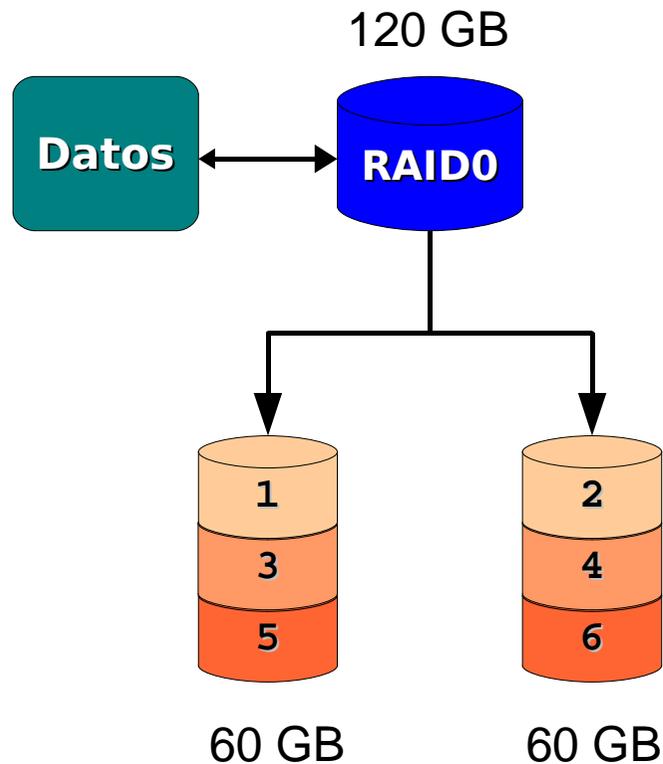
Niveles de RAID

Revisando conceptos

parity

- Esta técnica aplica a todo un grupo de dispositivos o a segmentos distribuidos a través de todo el grupo de dispositivos.
- En términos de RAID hablamos de dos tipos de paridad:
 - Paridad dedicada → Los datos de paridad de dos o más dispositivos son almacenados en un dispositivo adicional
 - Paridad distribuida → Los datos de paridad son distribuidos entre los dispositivos del arreglo.

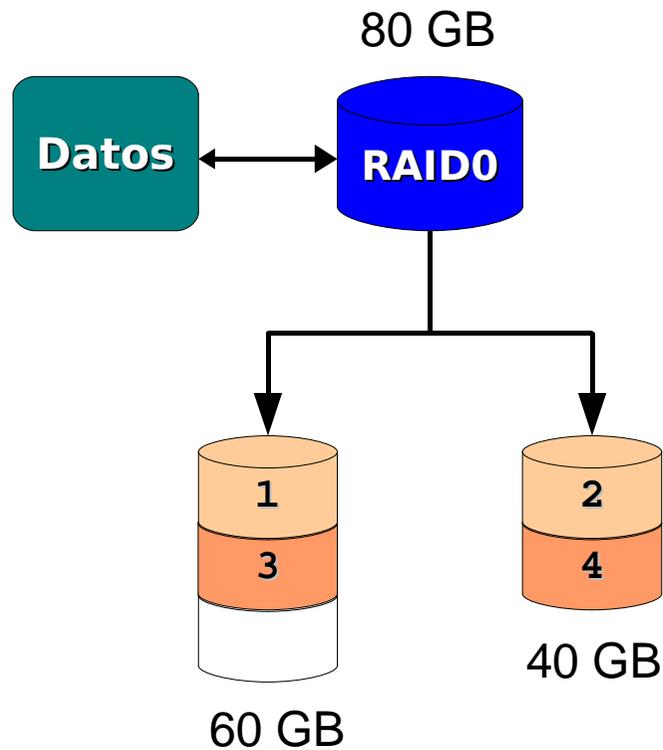
Niveles de RAID



RAID 0 (*disk striping*)

- Cada dispositivo del RAID se divide en segmentos de tamaño similar (ej. entre 8 KB a 1024 KB).
- Estos segmentos son intercalados de manera secuencial y repetida.
- El espacio de almacenamiento está compuesto por segmentos de todo el grupo de dispositivos.
- Ofrece un alto rendimiento ya que múltiples dispositivos son accedidos (lectura/escritura) simultáneamente.

Niveles de RAID



RAID 0 (*disk striping*)

- La cantidad total de almacenamiento es la suma de la capacidad de todos los dispositivos del grupo.
- Se pueden usar dispositivos de diferentes tamaños, recordando que el dispositivo de menor tamaño limita la cantidad de espacio usado en los demás dispositivos.



Un daño en cualquiera de los dispositivos hará inutilizable el RAID!!!

Niveles de RAID

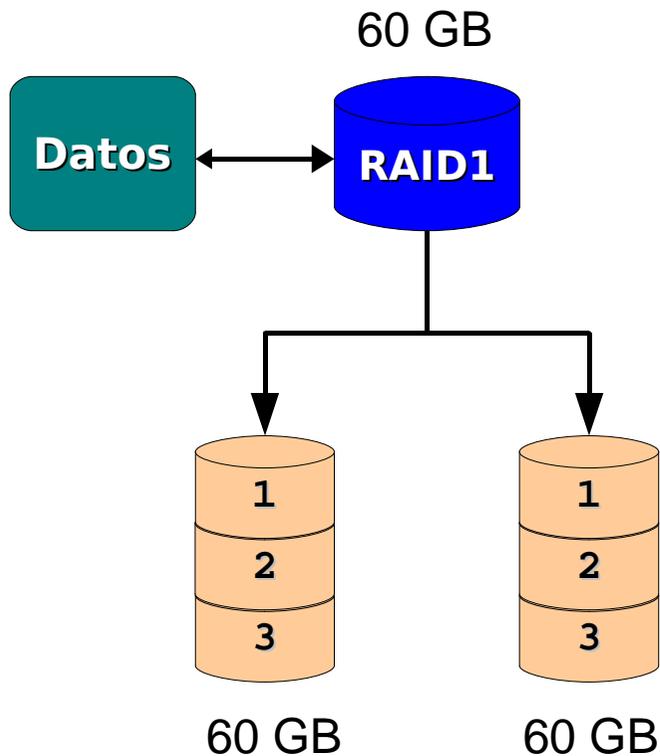
RAID 2 (*bit striping*)

- Divide los datos a nivel de *bits* distribuyendolos entre los dispositivos del arreglo.
- Hace uso de código *Hamming* para el chequeo de paridad.
- La segmentación a nivel de *bits* crea un alto impacto (lectura/escritura) en los recursos del sistema lo que lo hace inviable a nivel practico.

RAID 3 (*byte striping*)

- Divide los datos a nivel de *byte* distribuyendolos entre los dispositivos del arreglo.
- De forma similar que RAID 2 la segmentación de datos a nivel de *byte* crea un alto impacto (lectura/escritura) en los recursos del sistema haciendo inviable su implementación

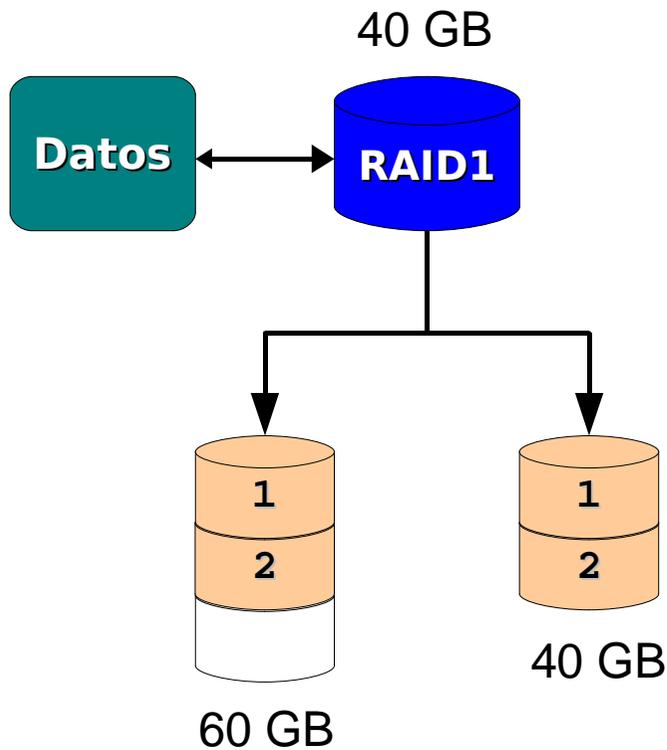
Niveles de RAID



RAID 1 (*disk mirroring*)

- Todos los datos escritos son duplicados (replica) en cada dispositivo del RAID.
- Según lo anterior ofrece 100% de redundancia.
- Un alto rendimiento ya que lo conforman múltiples dispositivos que pueden ser accedidos (lectura) mientras uno o más están ocupados.

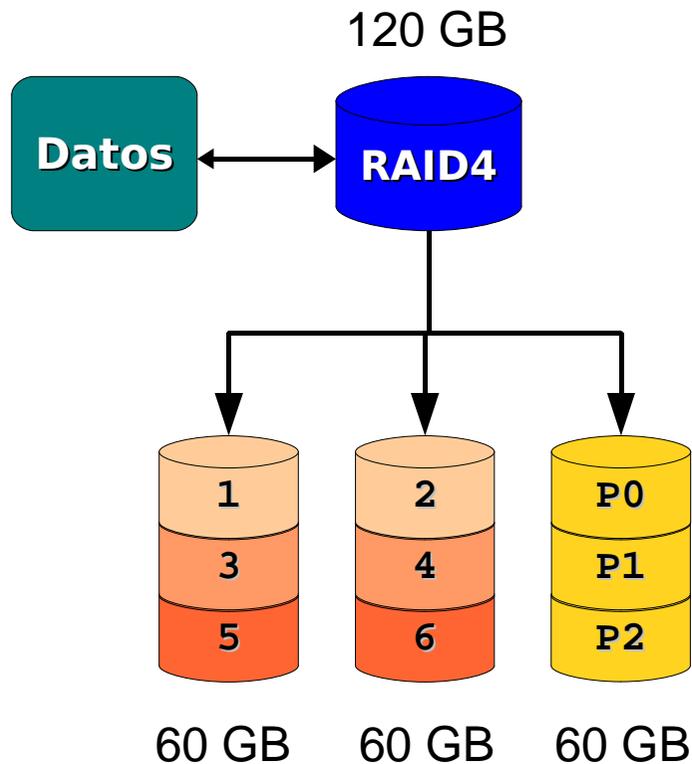
Niveles de RAID



RAID 1 (*disk mirroring*)

- La cantidad total de almacenamiento es igual al tamaño del dispositivo de menor capacidad.
- El uso de dispositivos de capacidad similar proporcionan un RAID optimo.
- Es un esquema de alto costo ya que cada dispositivo debe ser duplicado.

Niveles de RAID

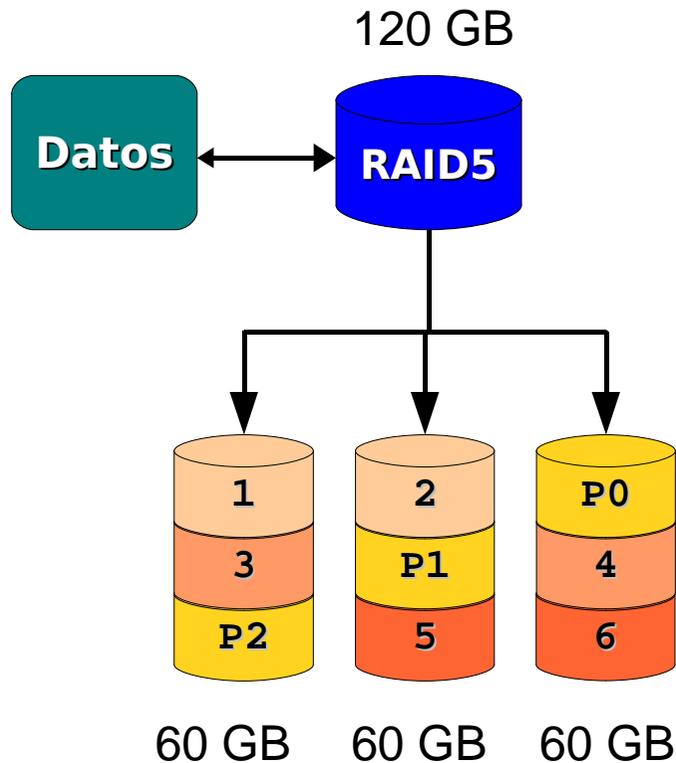


RAID 4

(block striping & dedicated parity)

- Segmenta los datos a nivel de bloques distribuyendolos entre los dispositivos del arreglo.
- Dedicar por completo uno de los dispositivos del arreglo para paridad.
- Es un esquema similar a RAID2 y RAID3 pero la división en bloques evita un alto impacto (lectura/escritura) en los recursos del sistema.

Niveles de RAID

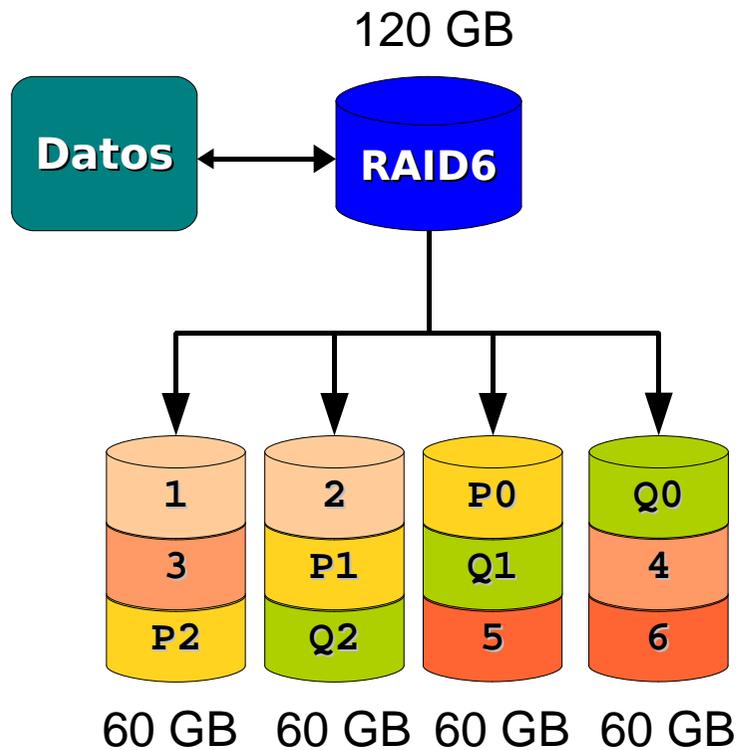


RAID 5

(block striping & distributed parity)

- Segmenta los datos a nivel de bloques distribuyendolos entre los dispositivos del arreglo de forma similar a RAID 4.
- Distribuye los datos de paridad entre todos los dispositivos del arreglo.
- RAID 4 y RAID 5 proveen redundancia ante la falla de un dispositivo en base a la información de paridad.

Niveles de RAID



RAID 6

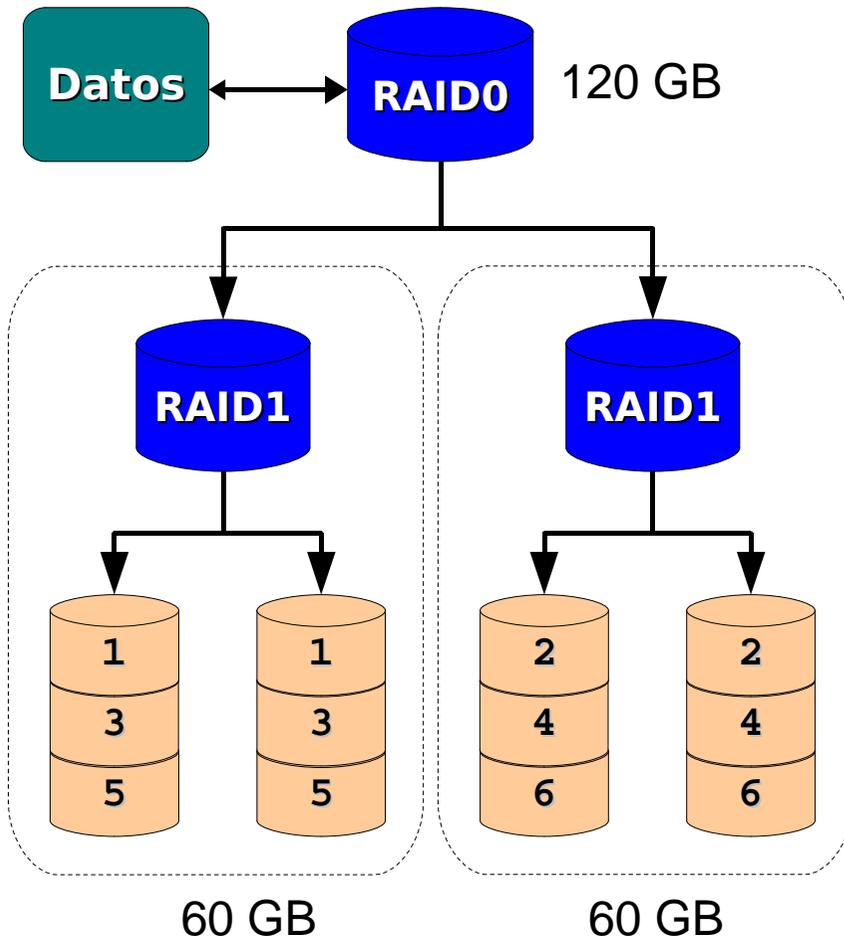
(block striping & distributed parity)

- Segmenta los datos a nivel de bloques distribuyendolos entre los dispositivos del arreglo de forma similar a RAID 4.
- Distribuye los datos de paridad entre todos los dispositivos del arreglo de forma similar a RAID 5.
- Un segundo conjunto de datos de paridad lo que provee redundancia ante la falla de dos dispositivos.

Niveles de RAID

- Es posible incrementar el rendimiento y la redundancia de un sistema de almacenamiento combinando diferentes niveles de RAID, estas combinaciones se conocen como arreglos híbridos.
- La mayoría de las tarjetas controladoras, gabinetes externos y *software* RAID soportan combinaciones de dos o más niveles.
- No todas las combinaciones soportadas y permitidas en *hardware* y *software* para RAID ofrecen beneficios.

Niveles de RAID

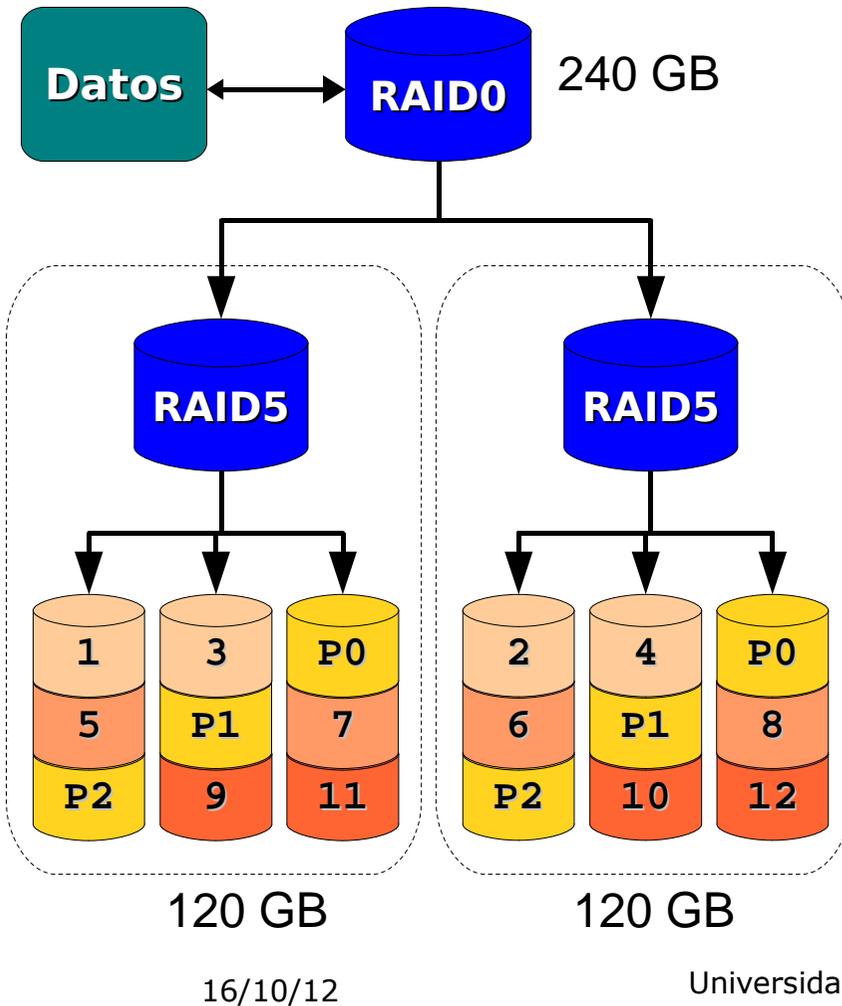


RAID 10

(*striping & mirror*)

- A pesar de su alto costo es muy utilizados.
- Combina el rendimiento (velocidad de acceso) de la segmentación de datos (*striping*) con las propiedades de la redundancia (*mirroring*).
- Si un dispositivo falla ambos lados del RAID 10 seguirán funcionando (aunque un lado en modo degradado)

Niveles de RAID

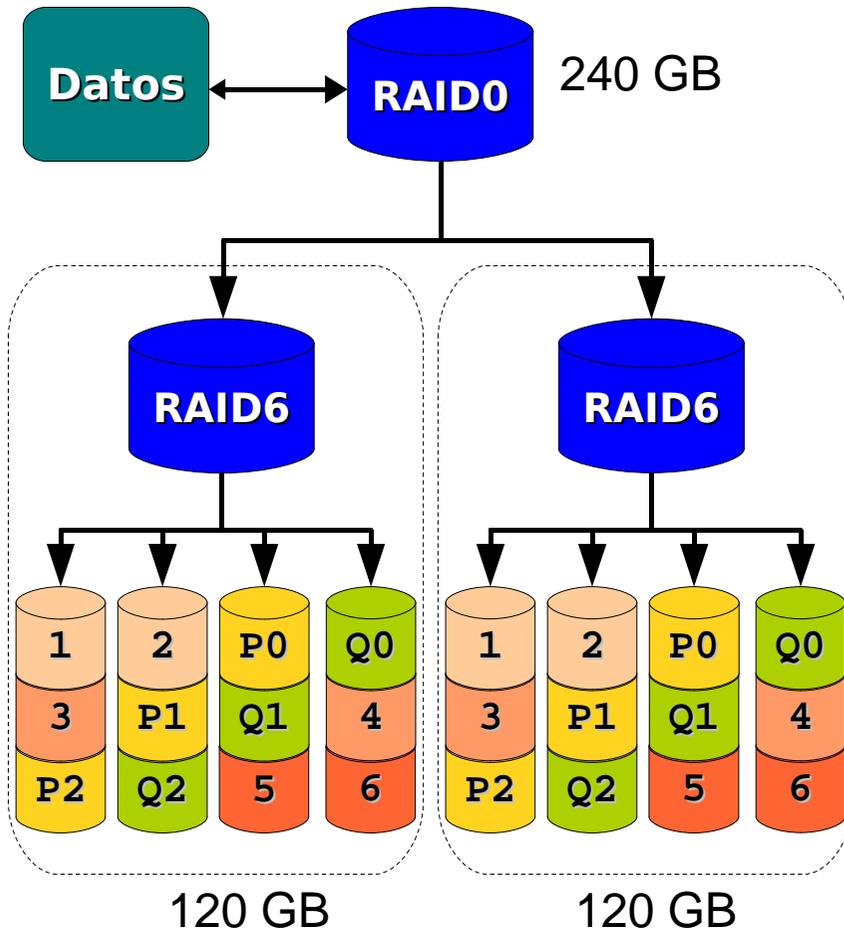


RAID 50

(*striping & parity*)

- En referencia a un RAID 10:
 - Menor costo.
 - Rendimiento de lectura más bajo (aunque sigue siendo bueno)
 - A nivel de escritura hay mayor rendimiento (velocidad).
- Cada RAID 5 puede soportar la falla de un dispositivo.

Niveles de RAID



RAID 60

(*striping & parity*)

- En referencia a un RAID 10:
 - Menor costo.
 - Rendimiento de lectura más bajo (aunque es bueno)
 - A nivel de escritura hay mayor rendimiento (velocidad).
- Cada RAID 6 puede soportar la falla de dos dispositivos (simultáneamente).

Comparando niveles

Nivel	Tipo	Dispositivos	Redundancia	Capacidad	Rendimiento (lectura)	Rendimiento (escritura)
RAID 0	<i>Striping (block level)</i>	$N > 1$	0	1	N	N
RAID 1	<i>Mirroring</i>	$N > 2$	$N - 1$	$1/N$	N	1
RAID 4	<i>Striping (block level) Parity (dedicated)</i>	$N > 2$	1	$1 - 1/N$	$N - 1$	$N - 1$
RAID 5	<i>Striping (block level) Parity (distributed)</i>	$N > 2$	1	$1 - 1/N$	$N - 1$	$N - 1$
RAID 6	<i>Striping (block level) Double Parity (distributed)</i>	$N > 3$	2	$1 - 2/N$	$N - 2$	$N - 2$

N = Cantidad de dispositivos en el grupo de almacenamiento

Comparando niveles

Nivel	Tasa de fallas (<i>fail rate</i>)	Posibles Aplicaciones
RAID 0	$1-(1-r)^N$	Almacenamiento de archivos grandes que no requieren redundancia en tiempo real.
RAID 1	r^N	Bases de datos y archivos de bajo contenido dinamico (poca capacidad)
RAID 4	$N(N-1)r^2$	Bases de datos, servidores de archivos, correo electronico, contenido.
RAID 5	$N(N-1)r^2$	Bases de datos, servidores de archivos, correo electronico, contenido.
RAID 6	$N(N-1)(N-2)r^3$	Bases de datos, servidores de archivos, correo electronico, contenido (mayor tolerancia a fallas que RAID 5)

N = Cantidad de dispositivos en el grupo de almacenamiento

r = % de error estimado por cada dispositivo del grupo de almacenamiento.

Tolerancia a fallas

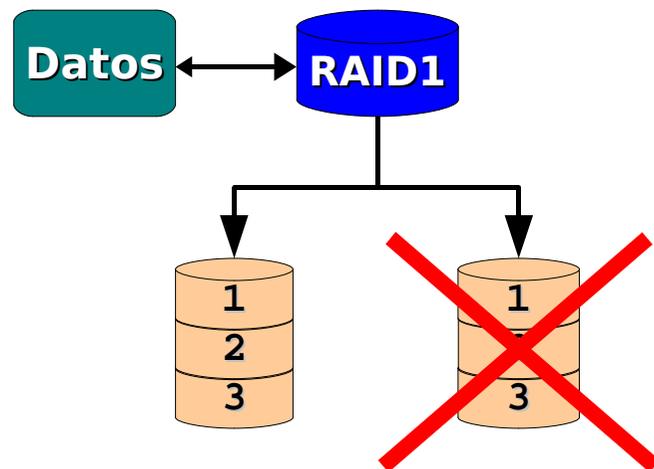
- Uno de los beneficios de RAID es la posibilidad de manejar fallas en los dispositivos sin detener el sistema de almacenamiento y sin intervención de un administrador (redundancia)
- Un RAID pre-configurado con los dispositivos necesarios, puede recuperarse de una falla por si mismo.
- En este sentido es importante revisar los siguientes conceptos:
 - Modo de degradación
 - *hot spares*
 - *hot swap*

Tolerancia a fallas

Revisando conceptos

Modo de degradación

- El momento en que un grupo de dispositivos falla por cualquier razón en un RAID (con redundancia)



Cuando ocurre:

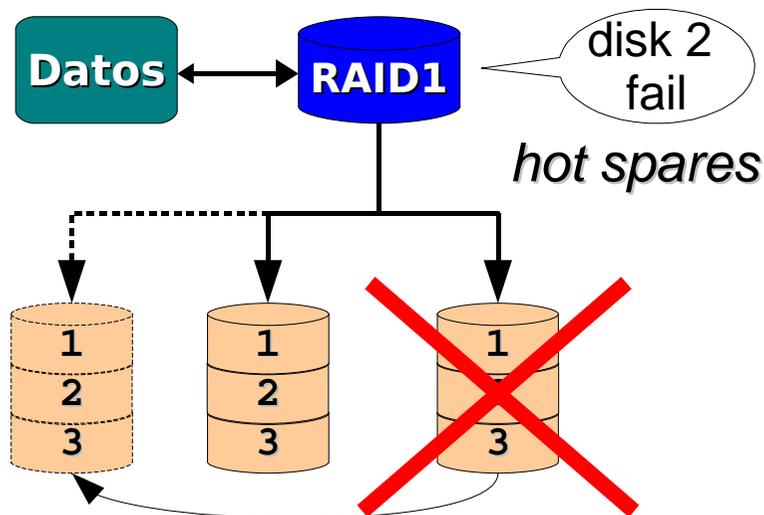
- Funcionamiento no óptimo.
- Redundancia comprometida.
- Ocurre en RAID 1, 5, 10, 50 y 60

Tolerancia a fallas

Revisando conceptos

hot spares

- Esta características en niveles con soporte de redundancia permite que RAID se recupere de una falla por si mismo.



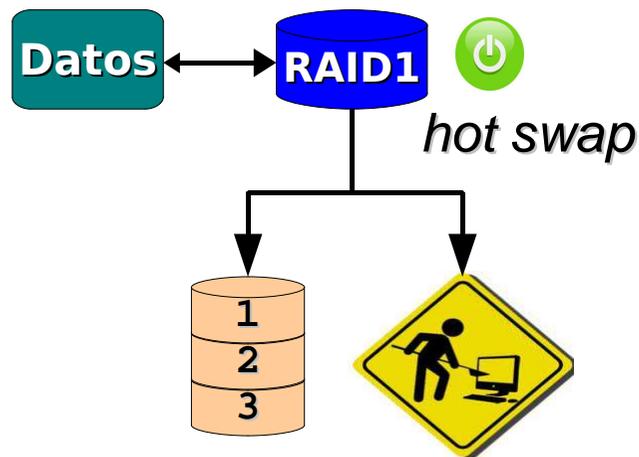
- Soporte en *hardware* y *software* RAID.
- Dispositivos extras en *stand by* esperando un falla para ocupar el lugar del dispositivo dañado.
- El RAID pasa a modo degradado mientras *hot spares* actua.

Tolerancia a fallas

Revisando conceptos

hot swap

- Esta característica en niveles con soporte de redundancia permite que en el RAID se puedan retirar dispositivos que han fallado con el sistema operando (*power on*)



Generalmente se usa en caso de:

- No se tiene espacio físico para instalar dispositivos con soporte *hot spares*
- Un dispositivo en modo *hot spares* está activo y se debe reemplazar el dispositivo dañado previendo fallas a futuro.

Confiabilidad y Disponibilidad

- Los sistemas RAID como cualquier otro puede presentar fallas que lo desvíen de su funcionamiento normal pero:
 - ¿Con que frecuencia ocurren estas fallas en el sistema?
 - En términos de tiempo ¿Como podemos medir la confiabilidad de un RAID ante posibles fallas?
- Revisemos de forma general el significado de algunos términos relacionados como:
 - *failure rate*
 - *mean time to data loss (MTTDL)*
 - *mean time to recovery (MTTR)*
 - *unrecoverable bit error rate (UBE)*

Confiabilidad y Disponibilidad

failure rate (Porcentaje de averías)

En termino generales se refiere a la frecuencia con la cual un sistema falla. En el contexto de RAID se aplican dos tipos de porcentajes de averías:

- Falla lógica → Si perdemos un solo dispositivo del RAID y su porcentaje esta dado por la suma del porcentaje de falla de cada dispositivo del grupo.
- Falla del sistema → Si perdemos datos en el arreglo, este porcentaje dependen del nivel de RAID implementado.



Falla lógica = Falla del sistema en un RAID sin redundancia!!!

Confiabilidad y Disponibilidad

mean time to data loss - MTTDL

(Tiempo promedio antes de la perdida de datos)

Tiempo promedio antes que la falla de uno o varios dispositivos pueda ocasionar perdidas de datos en un arreglo.

En conjunto con el tiempo promedio antes que ocurra una falla (*mean time to failure - MTTF*) son las dos métricas principales de confiabilidad en un arreglo.

- Si un arreglo no cuenta con redundancia $MTTDL = MTTF$
- Si aumenta la redundancia en un arreglo $MTTDL$ aumenta.
- $MTTDL \gg MTTF$ para un arreglo de alta confiabilidad.

Confiabilidad y Disponibilidad

mean time to recovery - MTTR

(Tiempo promedio de recuperación)

El tiempo que lleva recuperar un arreglo a su normal funcionamiento luego que ocurre una falla. Este tiempo incluiría:

- Tiempo en sustituir un dispositivo en falla.
- Tiempo para reconstruir el arreglo.
- En sistemas de alta disponibilidad el MTTR disminuye con el uso de arreglos (*hardware o software RAID*) que cuenten con soporte *hot spares* y/o *hot swap*

Confiabilidad y Disponibilidad

unrecoverable bit error rate - UBE

(Tasa de error de bit irrecuperable)

Relacionado con el tiempo en el cual un dispositivo de un arreglo no tiene capacidad para recuperar los datos después de aplicar en varios intentos códigos de redundancia (ej. *Codigos de Redundancia Ciclica - CRC*)

En referencia a un RAID 5 o RAID 6 la UBE puede comprometer la reconstrucción de un arreglo (con redundancia) que ha entrado en un modo de degradación.

Confiabilidad y Disponibilidad

- De forma general la disponibilidad de un sistema (en nuestro caso RAID) viene dada por la relación:

$$D = \frac{MTTF}{MTTF + MTTR}$$

- Algunos ejemplos recordando la notación de “nueves”:

Disponibilidad	Tiempo de apagado
90.1% (1 nueve)	36 dias/año
99% (2 nueves)	3.65 dias/año
99.9% (3 nueves)	8.76 hrs/año
99.99% (4 nueves)	52 min/año

Implementando en Linux

Verificando que dispositivos están instalados y como están distribuidos:

```
# fdisk /dev/sda
```

Usando herramientas de RAID en Linux (ej. Debian):

```
# apt-get install mdadm
```

Creando un arreglo:

```
# mdadm -C /dev/md0 -a yes -l 0 -n 2 /dev/sda10 /dev/sda11
```

Implementando en Linux

Dando formato a un arreglo:

```
# mkfs.ext3 /dev/md0
```

Definiendo en el arranque:

```
# vim /etc/fstab
```

```
    /dev/md0    /usr2    ext3    defaults    0    0
```

```
# reboot
```

Monitoreo de los arreglos:

```
# cat /proc/mdstat
```

```
# mdadm --detail /dev/md0
```

Implementando en Linux

Chequeando el arreglo:

- Simulando una falla en un disco:

```
# mdadm /dev/md0 -f /dev/sda1
```

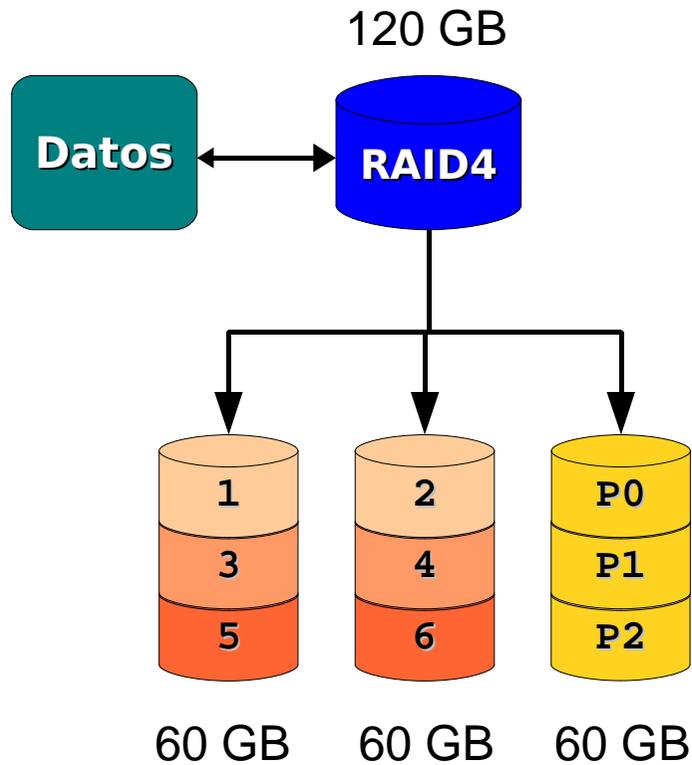
- Restableciendo una falla en un disco:

```
# mdadm /dev/md0 -r /dev/sda1
```

Mas documentación

```
# man mdadm
```

Apéndice A



De forma sencilla y general XOR:

Dispositivo1 -> 0 1 1 0 1 1 0 1

Dispositivo2 -> 1 1 0 1 0 1 0 0

Aplicamos XOR y almacenamos:

Dispositivo3 -> 1 0 1 1 1 0 0 1

Si fallara el dispositivo2:

Dispositivo3 -> 1 0 1 1 1 0 0 1

Dispositivo1 -> 0 1 1 0 1 1 0 1

Resulta -> 1 1 0 1 0 1 0 0

Dispositivo2

1 1 0 1 0 1 0 0

Referencias

- J. Ostergaard, *Software RAID HowTO*.
www.kernel.org
- *LVM & software RAID*.
GNS Systems
- D. Vadala, *Managing RAID on Linux*.
O'Reilly
- <http://en.wikipedia.org/wiki/RAID>