udp Escuela de Informática y Telecomunicaciones
FACULTAD DE INGENIERÍA

Almacenamiento

Adín Ramírez
adin.ramirez@mail.udp.cl

Sistemas Operativos (CIT2003-1)
1er Semestre 2015

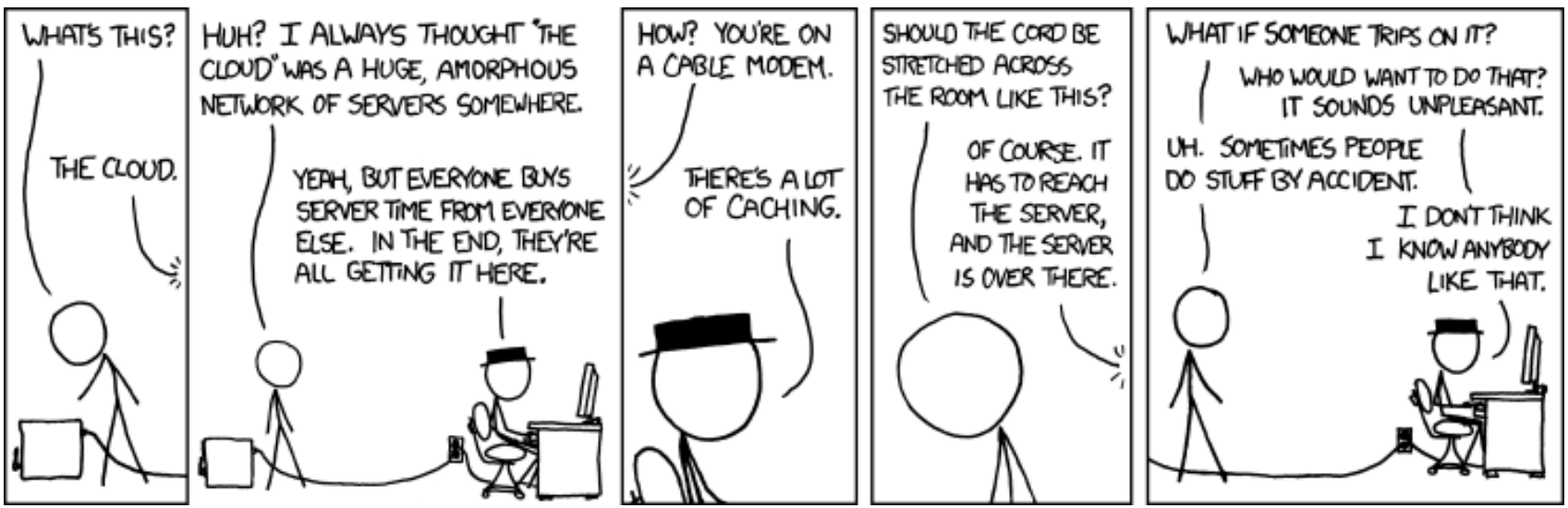
Adaptado de Dave Eckhardt

Puntos de hoy

- ¿Qué es almacenamiento?
- Anatomía de un disco duro
- Discos de estado sólido (SSD)
- Lo que deberían saber

Almacenamiento

- ¿Donde está la información cuando apagan su computadora?
- ¿Donde almacena la nube sus datos?



Almacenamiento

- Varios dispositivos
  - Cintas magnéticas
  - Discos duros
  - Disquetes (*floppy disk*)
  - CD-ROM
  - Memoria flash
  - MRAM
- ¿Qué tienen en común?
- ¿En qué se diferencian?

Características del almacenamiento

- No es volátil
  - Recuerda los datos sin electricidad
- Lenta (en comparación con la RAM)
  - Milisegundos o segundos en lugar de nanosegundos
  - No podemos ejecutar programas de estos dispositivos (tenemos que obtenerlos primero)
- Orientado a bloques
  - Obtener y almacenar grandes porciones de datos
    - Disquetes: 128/256/512 bytes
    - Discos: 512/4096 bytes
    - CD-ROM: 2048 bytes
    - Flash: 512/2048/4096 bytes (pero varían mucho)
  - Tiempo de obtener 1 byte = tiempo de obtener 1 bloque

Modelo de almacenamiento

- No volátil
  - Escribir, apagar, leer: debe de devolver el mismo valor
  - Independientemente del tiempo
- Espacio de direcciones
  - Los bloques tienen números
  - En tiempos antiguos: (C, H, S)
    - C, H, S son características geométricas de los discos antiguos
  - En los modernos: (LBA)
    - *Direcciones lógicas de bloques* corren de $0 \ldots N$

Escribir y leer

- `read_block(N)` $\Rightarrow$ bloque, sino error
  - A veces re intentar ayuda (pero no siempre)
- `write_block(N)` $\Rightarrow$ éxito, o error
  - Errores indican problemas "obvios"
  - Una escritura exitosa no garantiza una lectura posterior
  - Los dispositivos usualmente contienen un buffer
    - Una operación de escritura o completa o no tiene efecto
- Los dispositivos modernos soportan *tagged command queueing*
  - El sistema operativo genera múltiples solicitudes, cada una tiene una etiqueta (*tag*)
  - El dispositivo puede retornar los resultados en cualquier orden, con la etiqueta que envió el sistema operativo

Cola de comandos

- El disco realiza las solicitudes de lectura fuera de orden
  - Colas del SO: leer 37, leer 83, leer 2
    - El disco retorna 37, 2, 83
    - Y por eso compramos discos inteligentes y que encolan múltiples peticiones
  - Colas del SO: leer 37, leer 38, leer 39
    - El disco busca una vez, lee 37–40, además de 40–72 mientras está en el vecindario
    - Envía los sectores al SO conforme están disponibles
- El disto realiza las solicitudes de escritura fuera de orden también
  - Cola del SO: escribir 23, escribir 24, escribir 1000, leer 4–8
    - El disco escribe 24, 23, entrega 4, 5, 6, 7, 8, escribe 1000
    - ¿Qué pasa si falla el poder antes de la última solicitud de escritura?
    - ¿Qué pasa si falla el poder antes de las dos primeras solicitudes de escritura?
    - Conozcan sus I/O (interesante lectura)

¿Cómo asegura el sistema operativo la integridad de las estructuras de datos?

- Comandos especiales
- Escribir todos las solicitudes de escritura pendientes
  - Hace pensar al disco en “tener una barrera”
  - Puede enviar el *flush* a la cola de solicitudes
  - Puede aplicar a un conjunto de solicitudes pendientes antes del *flush*
  - Usado raramente por el sistema operativo
- Deshabilitar la escritura al cache
  - Hace pensar al disco “no sea tan moderno”

Ejemplos

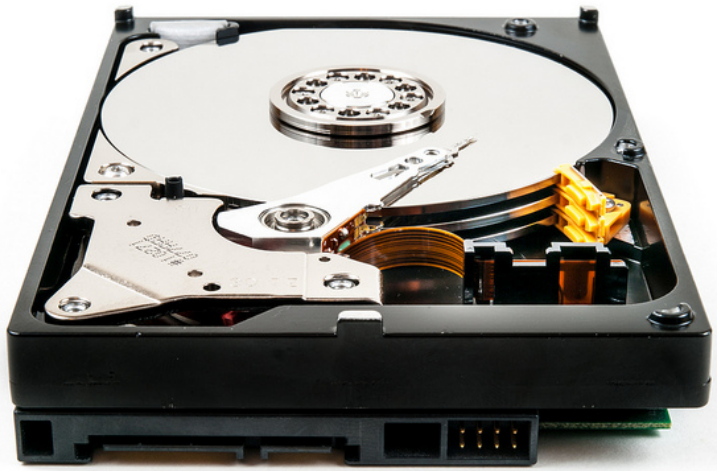
- Disco duro
  - Partes
  - Modelo de ejecución
- Memoria flash
  - Retos
    - Amplificación de escritura
    - Extensión de vida (*wear leveling*)

Anatomía del disco duro

Por fuera

http://en.wikipedia.org/wiki/Western_Digital

Por dentro



https://www.flickr.com/photos/wwarby/11644547564/in/photostream/

Anatomía del disco

- Un disco usualmente tiene múltiples discos, llamados platos
- Éstos giran a miles de RPM (5400, 7200, 10000, etc.)
- Discos más lentos utilizan menos poder
- Información es escrita hacia y leída desde los platos por las cabezas al final del brazo del disco

https://www.flickr.com/photos/wwarby/11644547564/in/photostream/

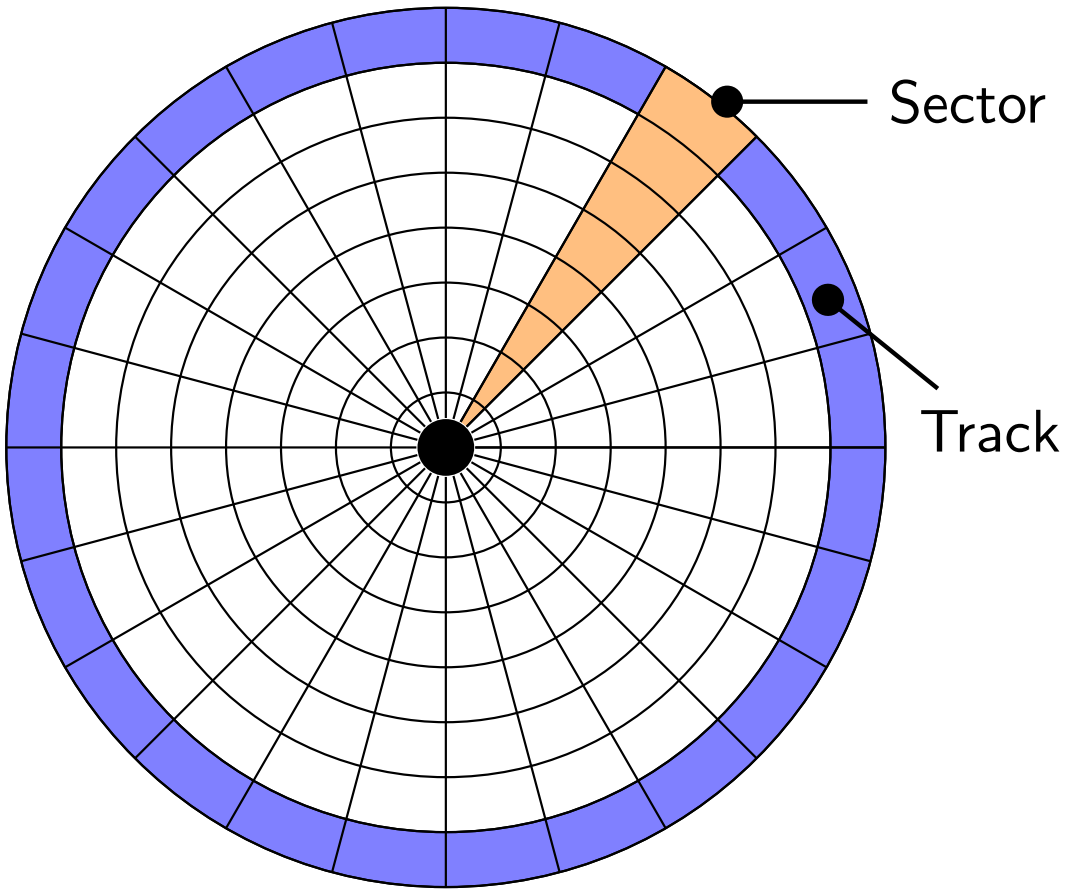
Anatomía del disco

- El brazo se mueve a través de un actuador
- Es lento
- Ambos lados de cada plato contienen información
- Cada lado es llamado una superficie
- Cada superficie tiene una cabeza de escritura/lectura

https://www.flickr.com/photos/wwarby/11644547564/in/photostream/

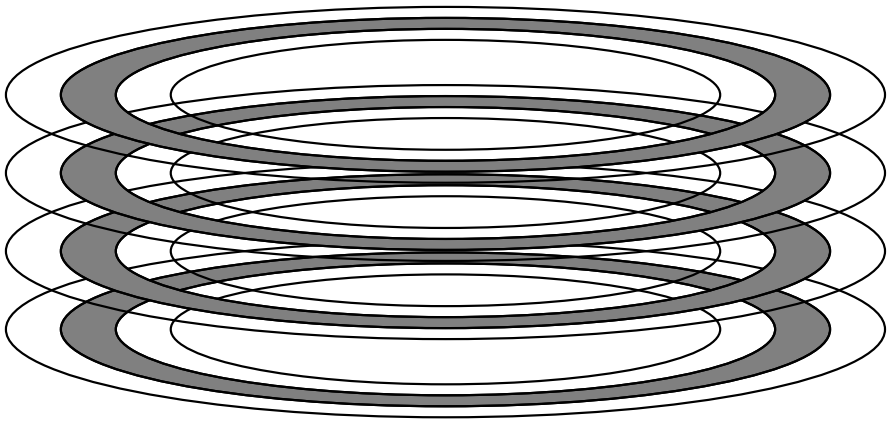
Anatomia de una superficie

- La superficie se divide en tracks
- Los tracks en sectores
- Un sector es la unidad más pequeña de transferencia de datos de y hacia el disco
  - 512 bytes —discos tradicionales
  - 2048 bytes —CD-ROMs
  - 4096 bytes —discos del 2010



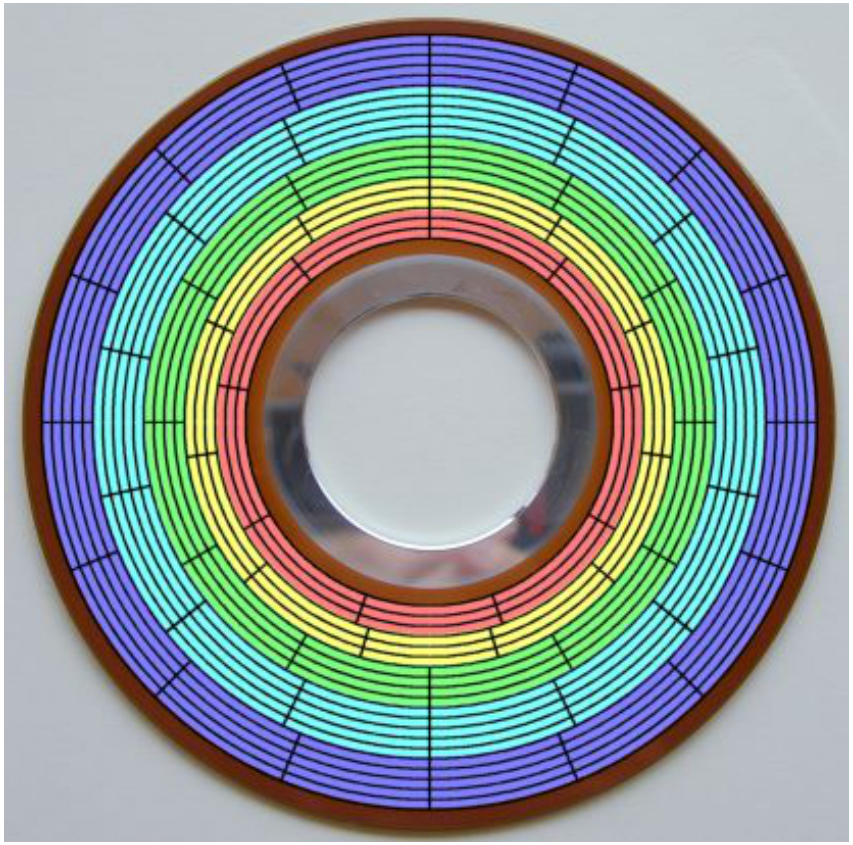
Cilindros

- Los mismos tracks en distintas superficies son llamados un cilindro
- Los accesos son (C, H, S)
  - C: cilindro
  - H: cabeza/superficie (head)
  - S: sector



Un disco real

- Los discos modernos graban a través de zonas de bits
  - El disco tiene mapas de # tracks a # sectores
  - Los sectores son del mismo tamaño lineal
  - Logical Block Address (LBA): la dirección del sector (similar como el número de página nombra un cuadro)



Lectura de un sector

- Debemos hacer dos cosas antes de transferir un sector
  - Mover la cabeza lectura/escritura al track apropiado (tiempo de búsqueda)
  - Esperar hasta que el sector buscado gire (retardo rotacional o latencia rotacional)
- Observemos
  - Los tiempo de búsqueda promedio son 2–10 mseg
  - Rotaciones de 5400/7200/10 K/15 K rpm equivalen a una demora rotacional de 11/8/6/4 mseg

Anatomía de un sector

- Encontrar un sector involucra trabajo real
  - Localizar el track correcto
  - Escanear las cabeceras del sector para encontrar el número
- Leer los datos
- Después, leer el código de verificación de error (*checksum*) y compararlo

Acceso dentro de un cilindro

Más rápido

- Las cabezas comparten un mismo brazo
  - Todas las cabezas están en el mismo cilindro simultáneamente
  - La cabeza activa está alineada, las otras están cerca
- Cambiar entre cabezas es “barato”
  - Desactivar una cabeza, y activar otra
  - Leer unos sectores, y alinear la cabeza para el nuevo sector
- Razón de transferencia óptima
  - Transferir todos los sectores en un track
  - Trasferir todos los tracks en un cilindro
  - Luego, movemos el brazo

Tiempo de acceso

- En promedio, debemos mover la cabeza de lectura/escritura sobre un tercio de los tracks
  - El tiempo para hacer esta operación es el "tiempo promedio de búsqueda"
  - 5400 rpm: $\sim$ 10 ms
  - 7200 rpm: $\sim$ 8.5 ms
- Además, debemos esperar media rotación, en promedio
  - El tiempo de hacer ésto es "demora promedio rotacional"
  - 5400 rpm: $\sim$ 5.5 ms
  - 7200 rpm: $\sim$ 4 ms
- Los números no encajan
  - Mientras el brazo se mueve, el disco gira también

Tiempo de acceso

- Tiempo total de acceso aleatorio es $\sim$ 7–20 milisegundos
  - 1000 ms/segundo, 20 ms/acceso $=$ 50 accesos/segundo
  - 50 transferencias de $\frac{1}{2}$ kilobyte por segundo $=$ 25 KByte/seg
  - Los discos son lentos
    - Pero, las transferencias de discos son de cientos de MBytes /seg
- Como programadores de SO, ¿qué podemos hacer al respecto?
  - Leer/escribir más por cada búsqueda (transferencias multi sector)
    - El cache del disco puede leer adelante, y retardar las escrituras
  - No buscar tan aleatoriamente
    - Colocar los datos relevantes cerca
    - Re ordenar las solicitudes
    - SO puede hacer "calendarización de disco" en lugar de FIFO
    - Históricamente relevante, más recientemente menos
    - Los discos también calendarizan internamente

Discos de estado sólido (SSD)

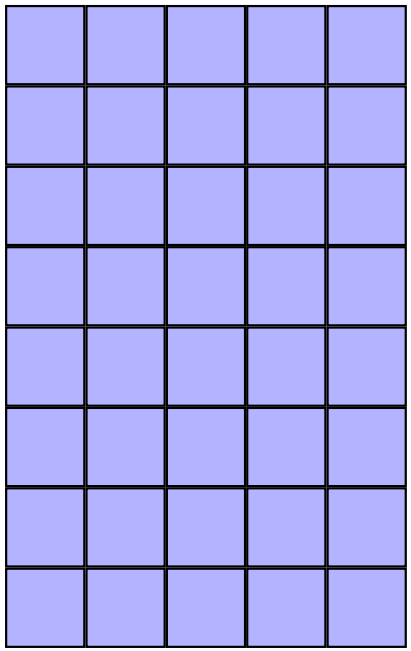
- ¿Qué es "estado sólido"?
  - Significado original: "no tubos al vacío"
  - Significado actual: "no partes en movimiento"
- ¿Qué es almacenamiento en "estado sólido"?
  - RAM respaldada por batería
    - Rápida
    - Permite servidores NFS completar escrituras RPCs sin tener que esperar por el disco
  - NOR flash
    - Accesible por palabra
    - Escrituras lentas, densidad baja
    - Se usa para arrancar dispositivos embebidos, configuración de almacenamiento
  - NAND flash
    - Lee/escribe páginas (512 B), borra bloques (16 KB)
    - La mayoría de SSD actuales son NAND flash
  - Y más cosas bajo desarrollo (Phase change memory, Magnetic RAM, Memristor memory)

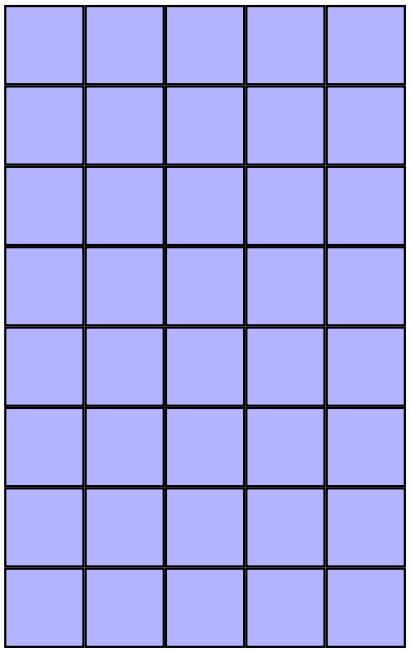
Características arquitecturales

- No hay partes que se muevan (mecánicas) así que no hay tiempo de búsqueda, o retardo rotacional
- Lecturas rápidas como las escrituras
- Escribir y borrar son distintas
  - Una página en blanco puede ser escrita (una sola vez)
  - Una página escrita debe ser borrada antes de re-escribir
  - Pero las páginas no se pueden borrar individualmente
    - Borrar funciona en bloques multí página (16 KB)
    - Borrar es muy lento
    - Borrar daña el bloque cada vez
- Implicaciones
  - Amplificación de escritura
  - Desgaste

Amplificación de escritura

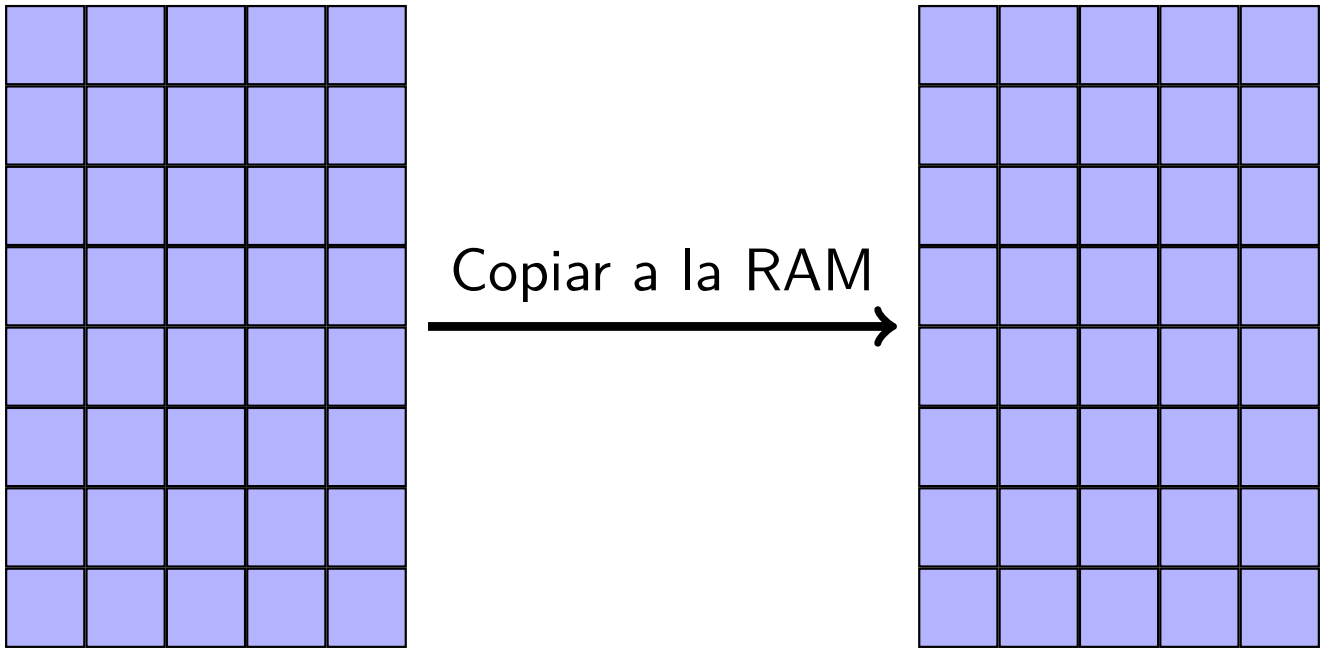
- Objetivo, copiar 2 páginas (1024 B) en un bloque (16 KB)





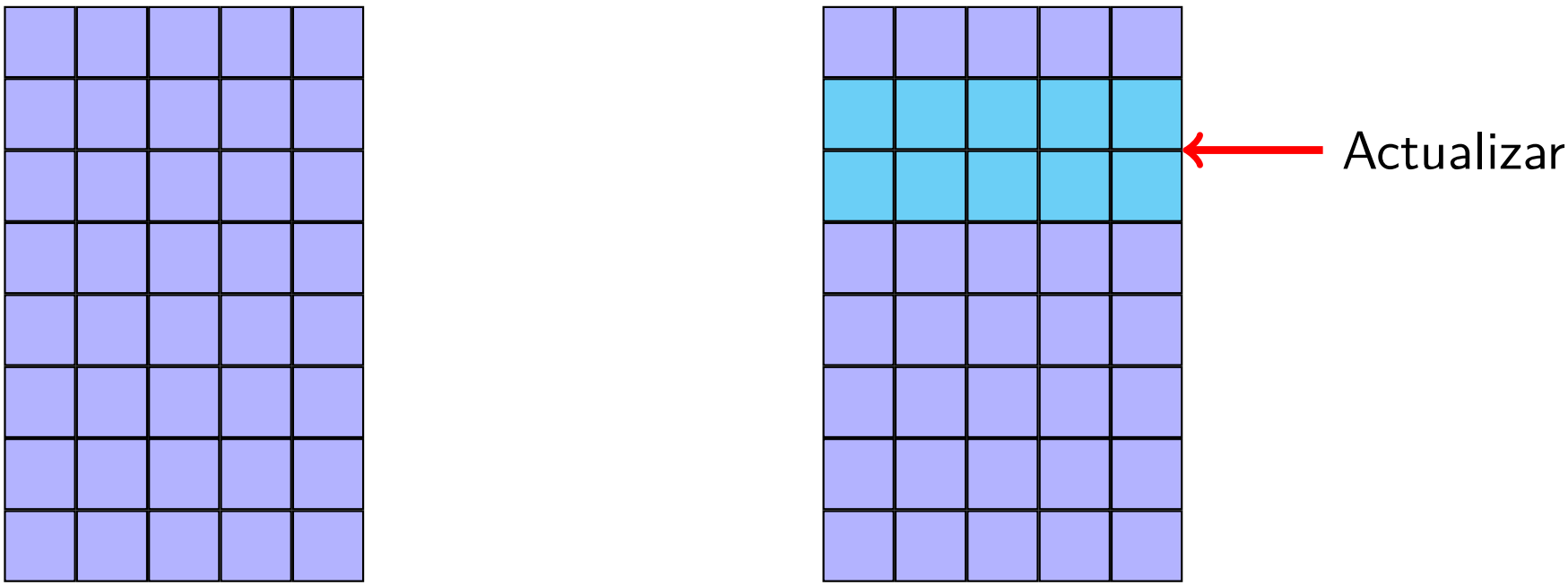
Amplificación de escritura

- Objetivo, copiar 2 páginas (1024 B) en un bloque (16 KB)



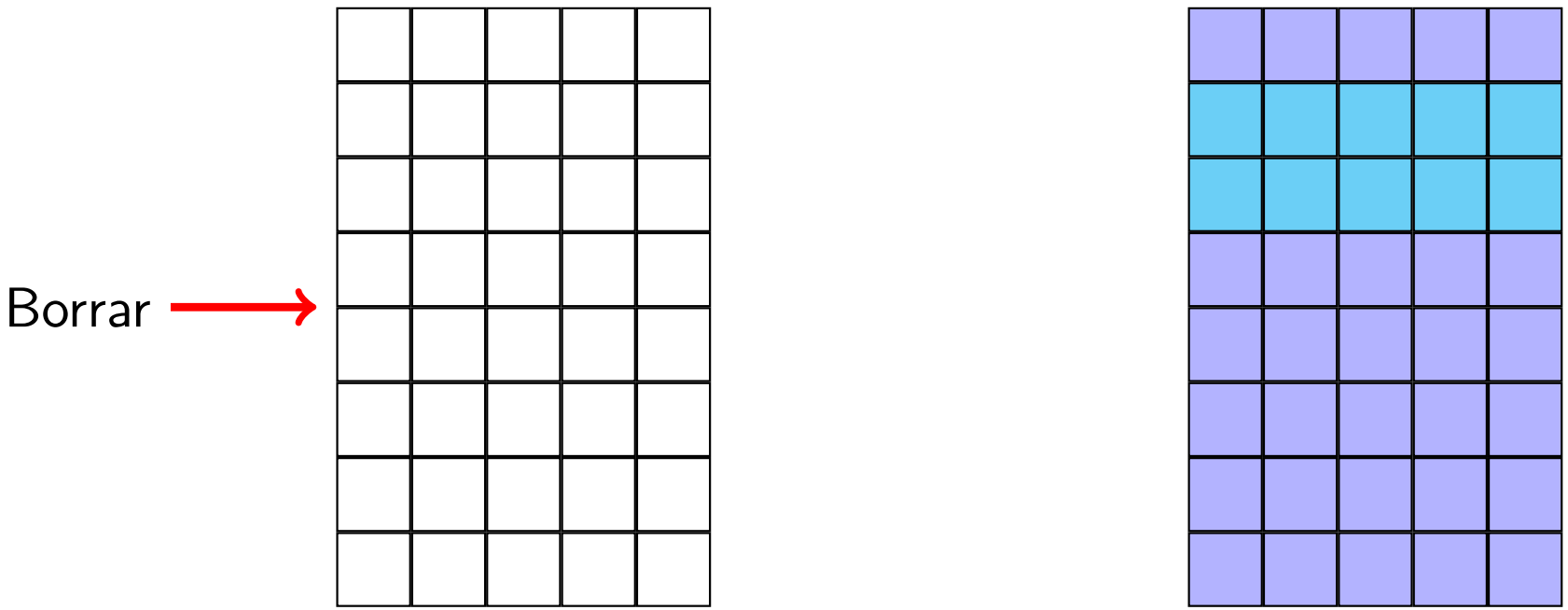
Amplificación de escritura

- Objetivo, copiar 2 páginas (1024 B) en un bloque (16 KB)



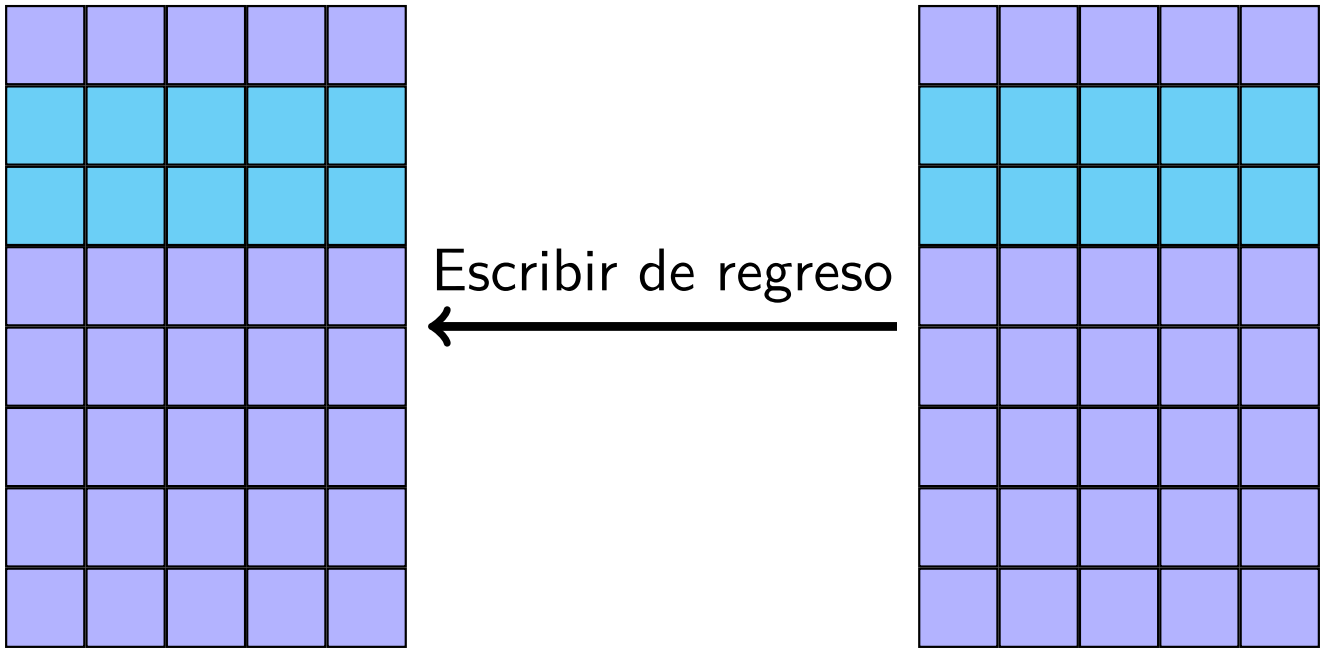
Amplificación de escritura

- Objetivo, copiar 2 páginas (1024 B) en un bloque (16 KB)



Amplificación de escritura

- Objetivo, copiar 2 páginas (1024 B) en un bloque (16 KB)



Resultado amplificación de escritura

- Lógicamente: escribimos 1 KB
- Físicamente: borramos y escribimos 16 KB
- Factor de amplificación: 8

Desgaste (*Wear leveling*)

- El mal caso
  - El sistema de archivos escribe en el mismo bloque repetidamente
  - Borrar daña parte de la flash
  - $\sim$ 10000 borradas destruye un bloque
- Estrategia: mentirle al sistema operativo
  - El anfitrión cree que está escribiendo en un bloque específico (LBA)
  - Almacenar la información en otra parte
    - De manera secreta rempaeamos las direcciones del anfitrión a las direcciones NAND
    - FTL —flash translation layer
  - Cada parte del disco se mueve hacia otra parte de la flash con el tiempo
  - Sobre provisión
    - Prometer menos espacio del que existe
    - El espacio extra se utiliza para reemplazar los bloques destruidos
  - Usar la sobre provisión conforme se destruyen los bloques

Resumen SSD

- SSD vs. disco
  - SSD implementa un modelo de disco regular
    - Sectores LBA
    - Escribir sector, leer sector, parquear cabezas, etc.
  - Las operaciones de lectura son extremadamente rápidas (100 veces más rápidas), no hay tiempo de búsqueda o retraso rotacional (cada sector es cercano)
  - Las operaciones de escritura varían (tal vez 100 veces más rápido, tal vez no hay aumento de velocidad)
  - SSD usan menos energía que los discos
  - SSD son resistentes a shocks
  - Escribir en la SSD deteriora el disco más rápido
  - SSD son más caros

Resumen SSD

- Oportunidades y amenazas
  - El comando `TRIM` agiliza las escrituras
  - Borrar el disco de manera segura puede no ser posible
- El futuro
  - Más SSD
  - Más discos también
  - Sistemas híbridos que tomen ventaja de cada característica

¿Qué recordar?

- El almacenamiento es lento
  - Lo que hagamos toma milisegundos
- El almacenamiento miente
  - Obtenemos un número de bloques de disco
  - No hay manera de conocer donde están en el disco
  - LBA es una aproximación de cercanía
- Modelo de fallo
  - A veces una lectura falla
  - Escribir en ese bloque causará que el dispositivo re-mapee (discos y SSD)
  - Cuando el espacio re-mapeado se termine, el dispositivo se rehusará a escribir
- Seguridad
  - El borrar información de la flash es incierto
  - Sugerencia: encriptar

Lecturas extra

- Reliably Erasing Data from Flash-based Solid State Drives, Wei et al., UCSD, FAST '11, http://www.usenix.org/legacy/events/fast11/tech/full_papers/Wei.pdf
- A Conversation with Jim Gray Dave Patterson, ACM Queue, June 2003, http://queue.acm.org/detail.cfm?id=864078
- Terabyte Territory, Brian Hayes, American Scientist, May/June 2002, http://www.americanscientist.org/issues/pub/terabyte-territory