Sistemas Operacionais

Gestão de entrada/saída - armazenamento

Prof. Carlos Maziero

DInf UFPR, Curitiba PR

Fevereiro de 2025

Conteúdo

1 Discos rígidos

2 Escalonamento de disco

3 Sistemas RAID

4 Dispositivos de estado sólido

5 Interfaces de acesso

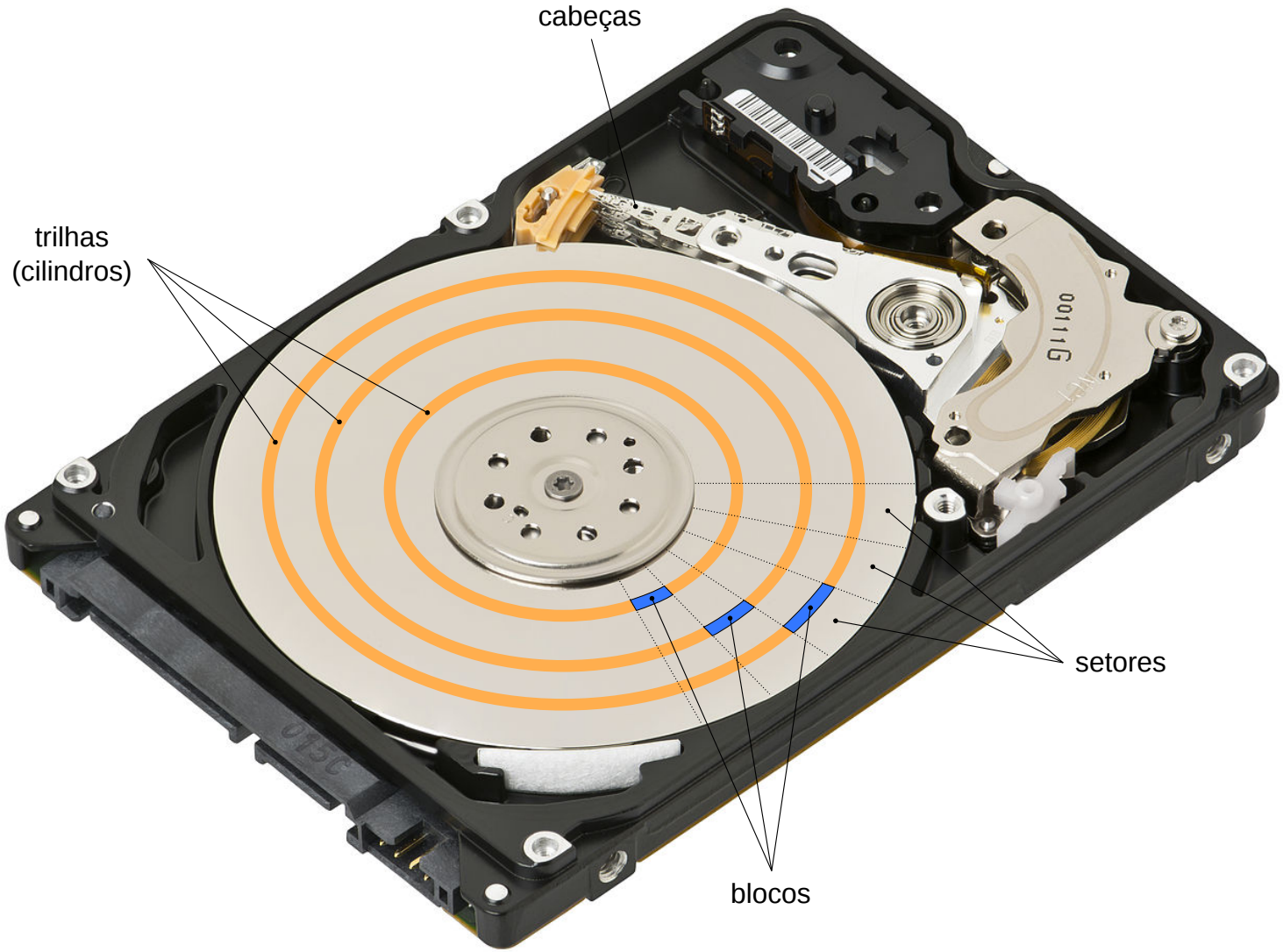
Discos rígidos

Dispositivo de armazenamento **magnético**

Características:

- Criado em 1954
- Um ou mais discos metálicos
- Velocidade de rotação entre 4.200 e 15.000 RPM
- Capacidade entre 100s GB e 10 TB
- Taxa de transferência entre 0.5 e 2 Gbps
- Latência entre 2 e 10 ms

Estrutura física



Estrutura lógica do disco rígido

Estrutura:

- Faces (ou cabeças): duas por disco metálico
- Trilhas (ou cilindros): faixas concêntricas
- Setores: “fatias” angulares

Blocos físicos:

- Interseção entre cabeça, trilha e setor
- Tamanho fixo de 512 ou 4.096 bytes

Endereçamento dos blocos:

- Esquema CHS: *Cylinder, Head, Sector* (interno)
- Esquema LBA: *Large Block Array* (*firmware* ou BIOS)

Escalonamento de acessos

O disco é um dispositivo lento!

- Latência rotacional $t_r \approx 5ms$
- Tempo de busca $t_s \approx 10ms$ (*seek time*)

O disco é um dispositivo sequencial: trata um pedido por vez!

Tratamento dos pedidos de acesso ao disco:

- Pedidos dos processos são mantidos em uma fila
- A fila é organizada de acordo com um algoritmo
- Busca-se **desempenho** e **justiça**

Algoritmos de escalonamento clássicos

- FCFS - *First Come, First Served*
- SSTF - *Shortest Seek-Time First*
- SCAN, C-SCAN, LOOK e C-LOOK ("elevador")

Exemplo: fila de pedidos de acesso aos blocos:

278, 914, 447, 71, 161, 659, 335

Cabeça do disco se encontra no bloco 500

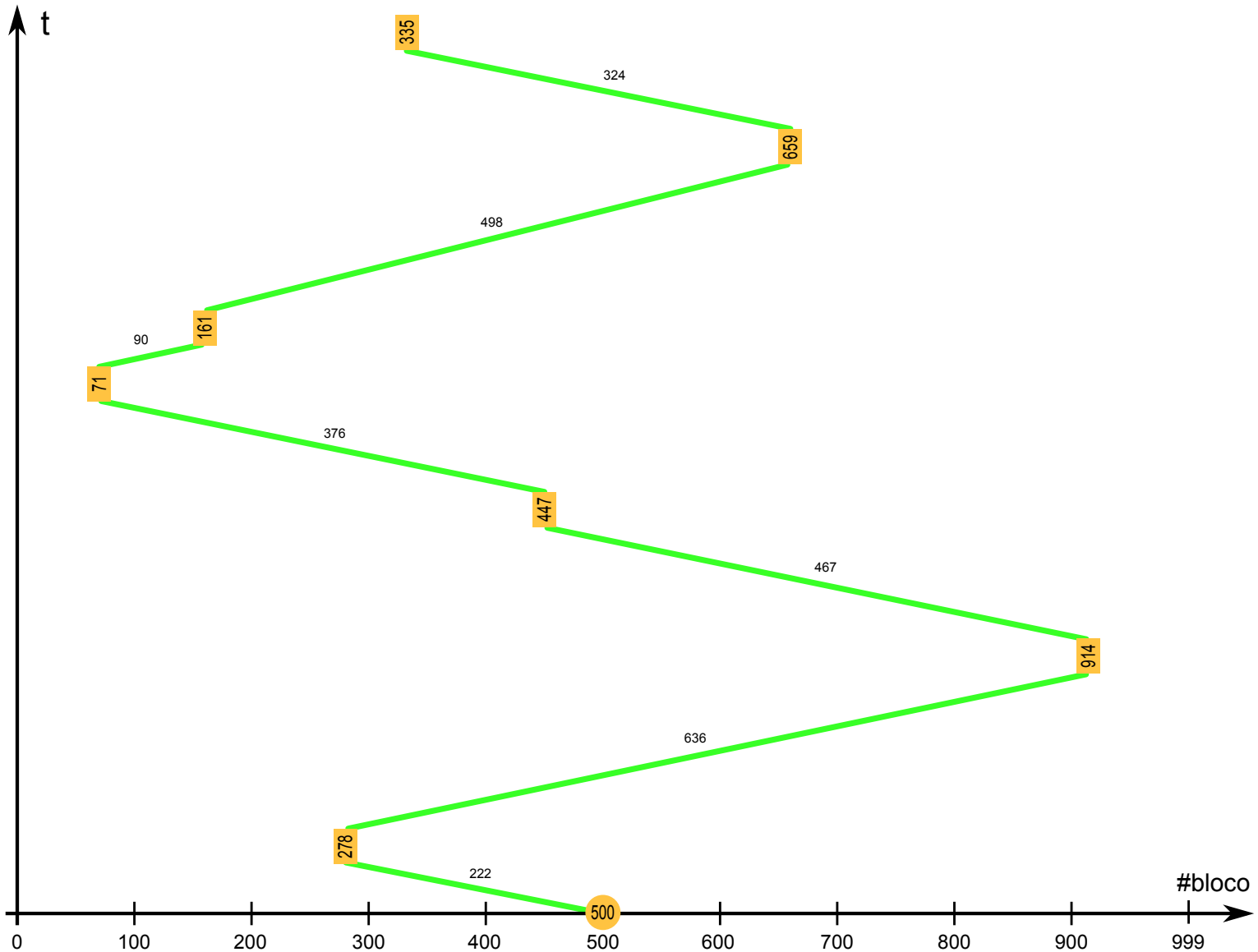
Escalonamento FCFS

Atender as requisições **na ordem** em que foram emitidas.

$$500 \xrightarrow{222} 278 \xrightarrow{636} 914 \xrightarrow{467} 447 \xrightarrow{376} 71 \xrightarrow{90} 161 \xrightarrow{498} 659 \xrightarrow{324} 335$$

Deslocamento da cabeça: 2.613 blocos

Escalonamento FCFS



Escalonamento SSTF

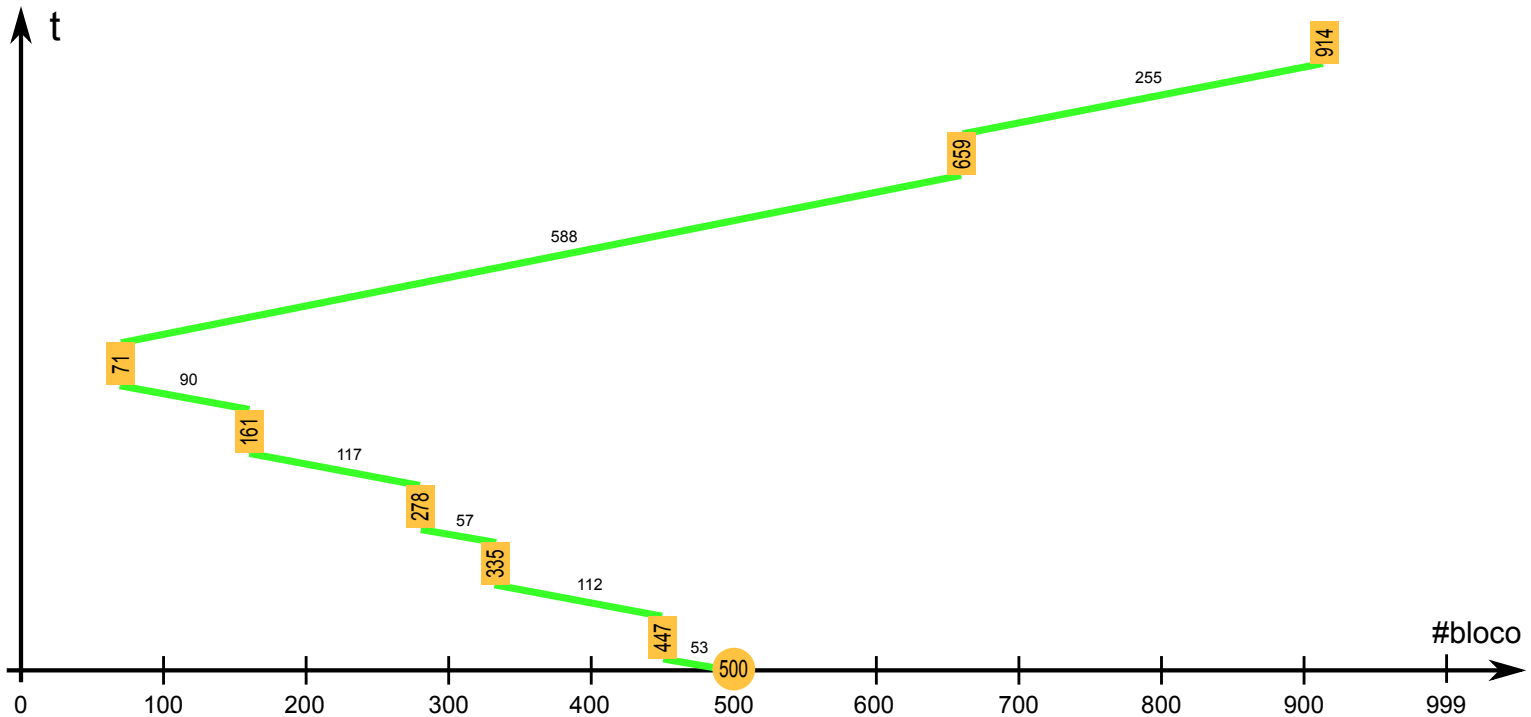
Shortest Seek Time First: menor tempo de busca primeiro.

Atender o pedido que está **mais próximo** da cabeça.

$$500 \xrightarrow{53} 447 \xrightarrow{112} 335 \xrightarrow{57} 278 \xrightarrow{117} 161 \xrightarrow{90} 71 \xrightarrow{588} 659 \xrightarrow{255} 914$$

Deslocamento da cabeça: 1.272 blocos

Escalonamento SSTF



Risco de inanição (*starvation*) para blocos distantes

Escalonamento SCAN

A cabeça “varre” (*scan*) o disco nos dois sentidos

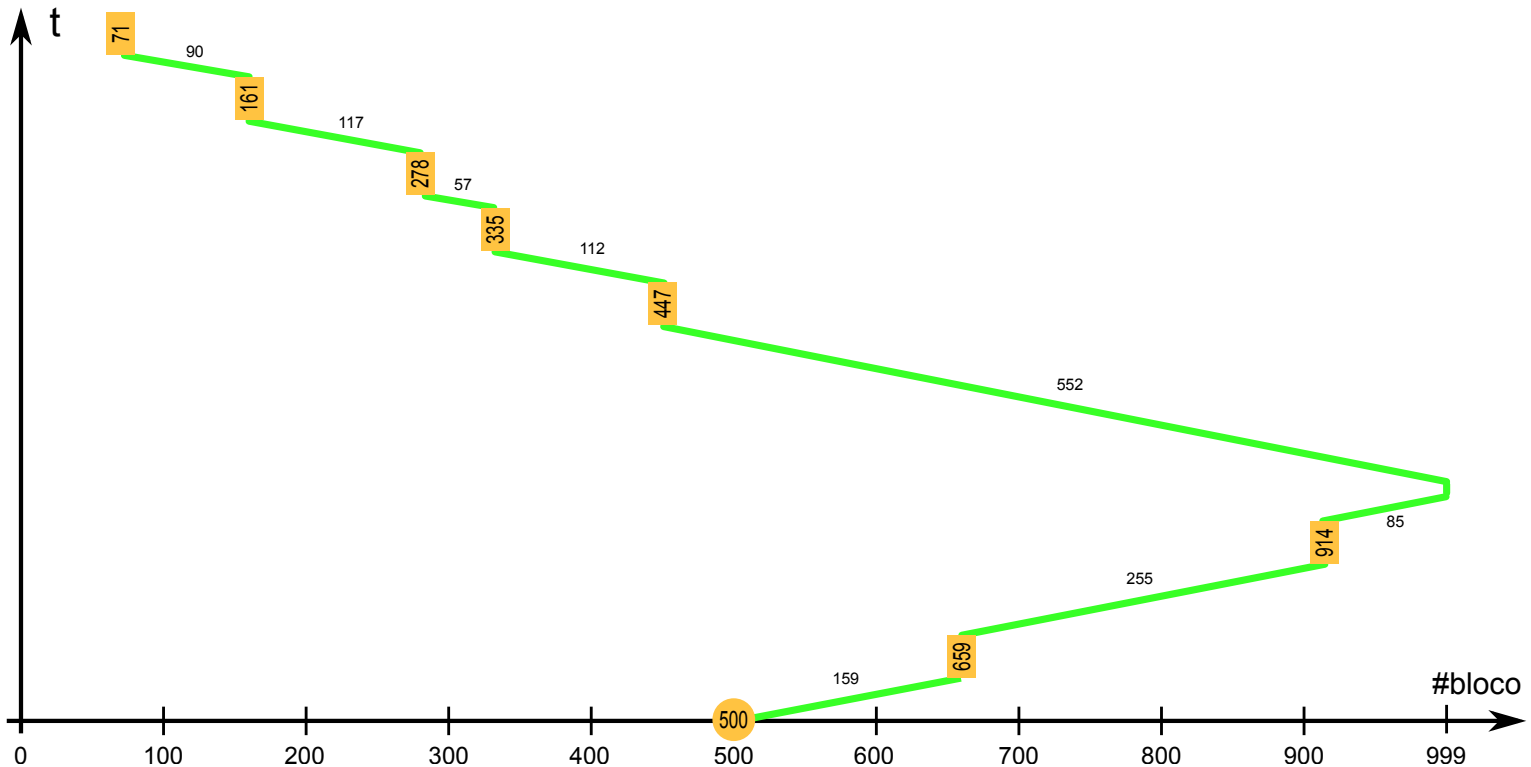
Também chamado de “algoritmo do elevador”

Bom desempenho e mais justiça no atendimento dos pedidos

$$500 \xrightarrow{159} 659 \xrightarrow{255} 914 \xrightarrow{85} 999 \xrightarrow{552} 447 \xrightarrow{112}$$
$$\longrightarrow 335 \xrightarrow{57} 278 \xrightarrow{117} 161 \xrightarrow{90} 71$$

Deslocamento da cabeça: 1.427 blocos

Escalonamento SCAN



Escalonamento C-SCAN

Variante “circular” do algoritmo SCAN.

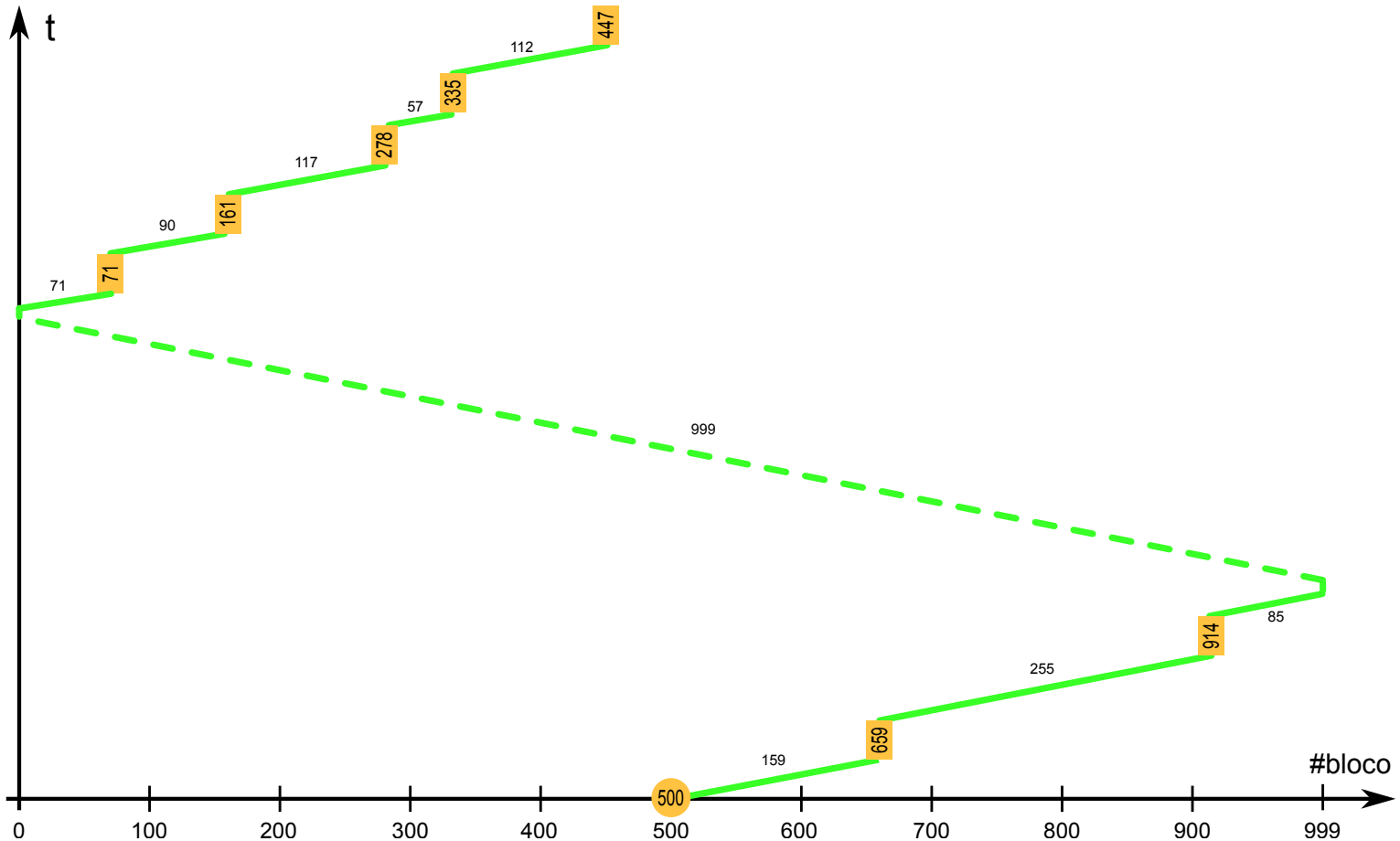
Varre o disco somente em um sentido.

Tempo de espera mais homogêneo aos pedidos pendentes.

$$500 \xrightarrow{159} 659 \xrightarrow{255} 914 \xrightarrow{85} 999 \xrightarrow{999} 0 \xrightarrow{71}$$
$$\longrightarrow 71 \xrightarrow{90} 161 \xrightarrow{117} 278 \xrightarrow{57} 335 \xrightarrow{112} 447$$

Deslocamento da cabeça: 1.945 blocos.

Escalonamento C-SCAN



Escalonador LOOK

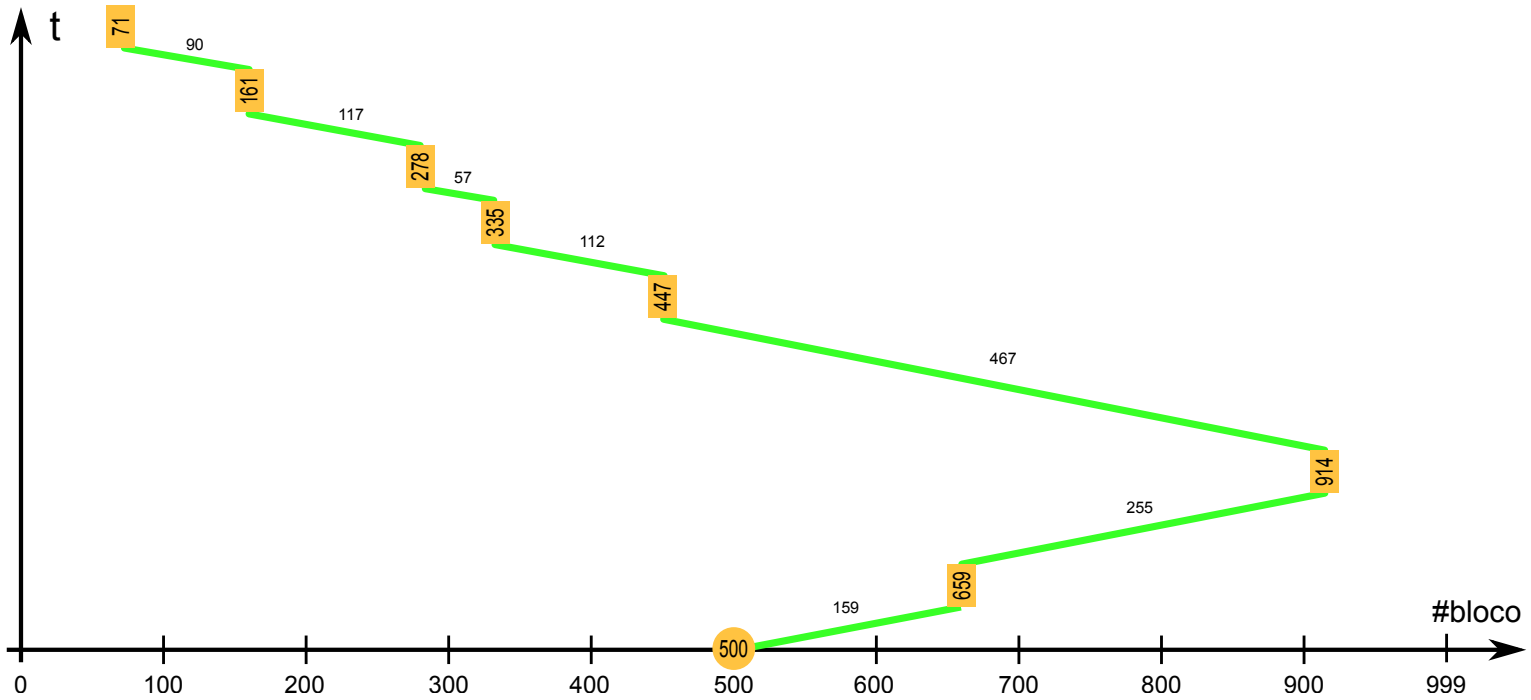
Otimização do algoritmo SCAN

A cabeça não avança até o final do disco

$$500 \xrightarrow{159} 659 \xrightarrow{255} 914 \xrightarrow{467} 447 \xrightarrow{112} 335 \xrightarrow{57} 278 \xrightarrow{117} 161 \xrightarrow{90} 71$$

Deslocamento da cabeça: 1.257 blocos

Escalonamento LOOK



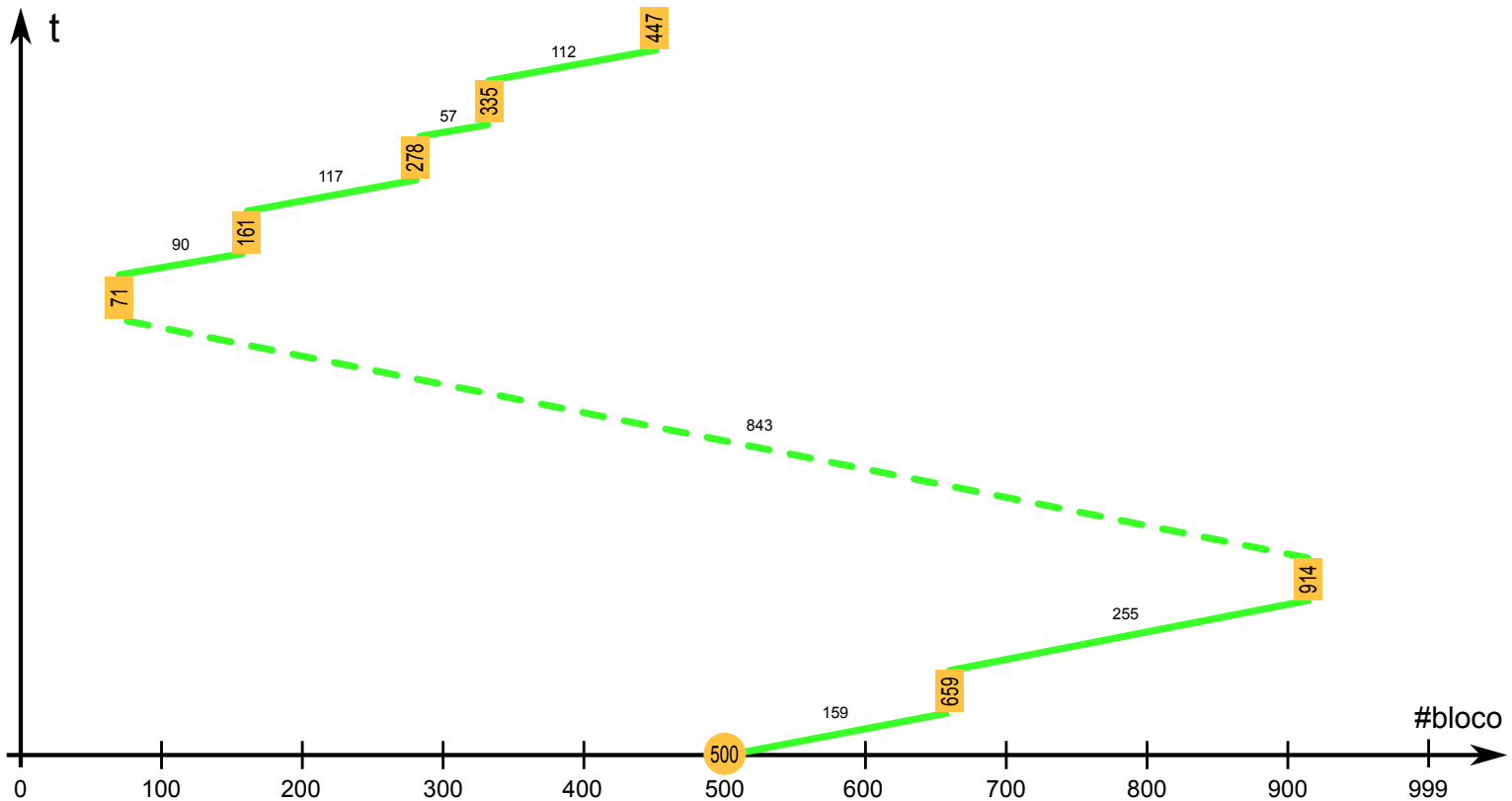
Escalonador C-LOOK

Otimização do algoritmo C-SCAN

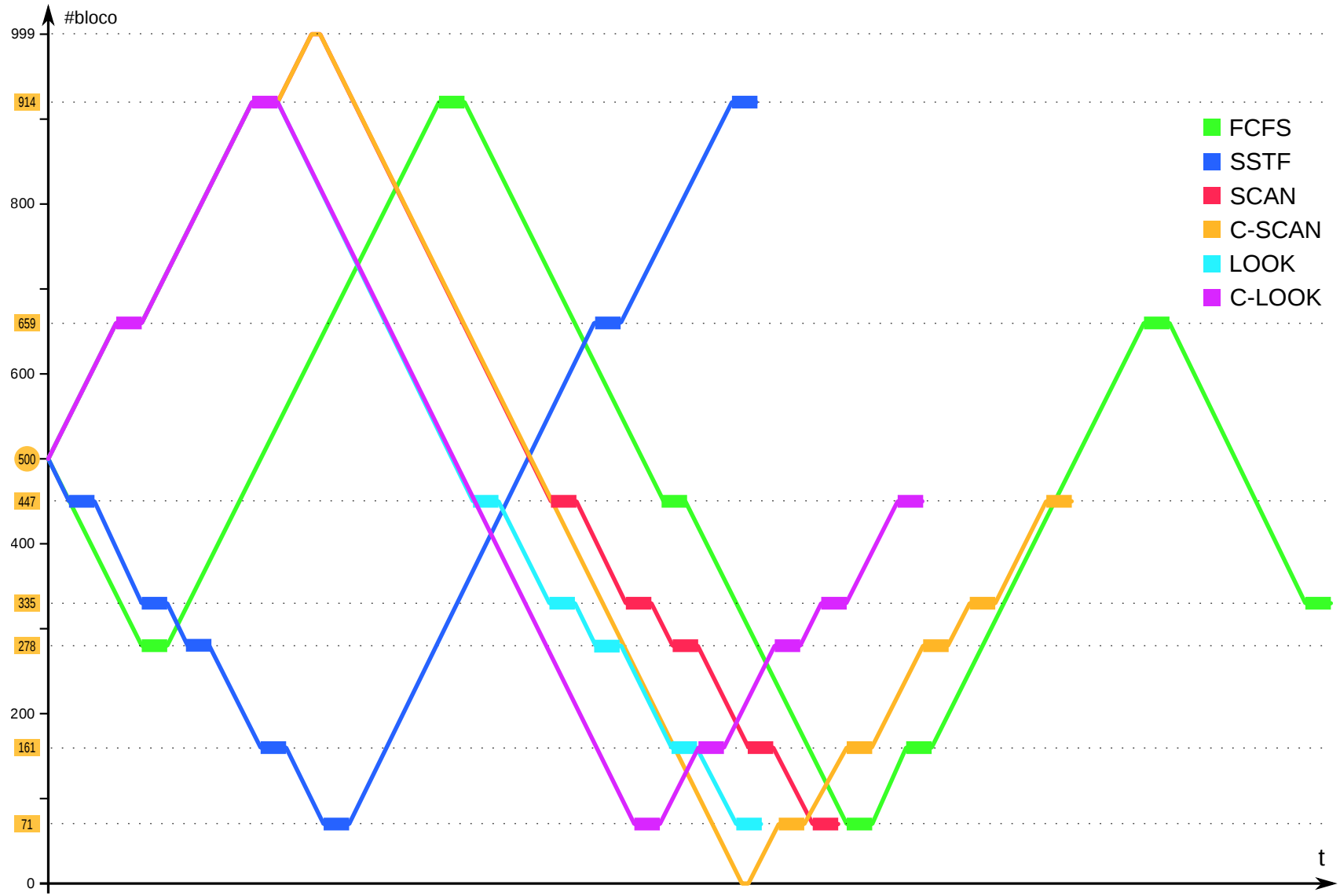
$$500 \xrightarrow{159} 659 \xrightarrow{255} 914 \xrightarrow{843} 71 \xrightarrow{90} 161 \xrightarrow{117} 278 \xrightarrow{57} 335 \xrightarrow{112} 447$$

Deslocamento da cabeça: 1.644 blocos

Escalonamento C-LOOK



Comparativo de escalonadores



Escalonadores de disco no Linux

- **Noop**
  - Baseado em FCFS
  - Agrupa pedidos ao mesmo bloco ou blocos adjacentes
  - Usado em SSD e sistemas RAID
- **Deadline** e **Anticipatory**
  - Associa prazos aos pedidos (500 ms leitura, 5s escrita)
  - Baseado no algoritmo C-SCAN, priorizando os prazos
  - *Anticipatory*: agrupa leituras do mesmo processo
- **CFQ - Completely fair queueing**
  - Pedidos são distribuídos em várias filas (64 por default)
  - Cada fila tem uma fatia de tempo para acessar o disco

Consultar `/sys/block/[device]/queue/scheduler`

Sistemas RAID

Problemas dos discos rígidos:

- Discos são lentos
- Discos podem falhar, levando à perda de dados

Estratégia RAID: *Redundant Array of Independent Disks*

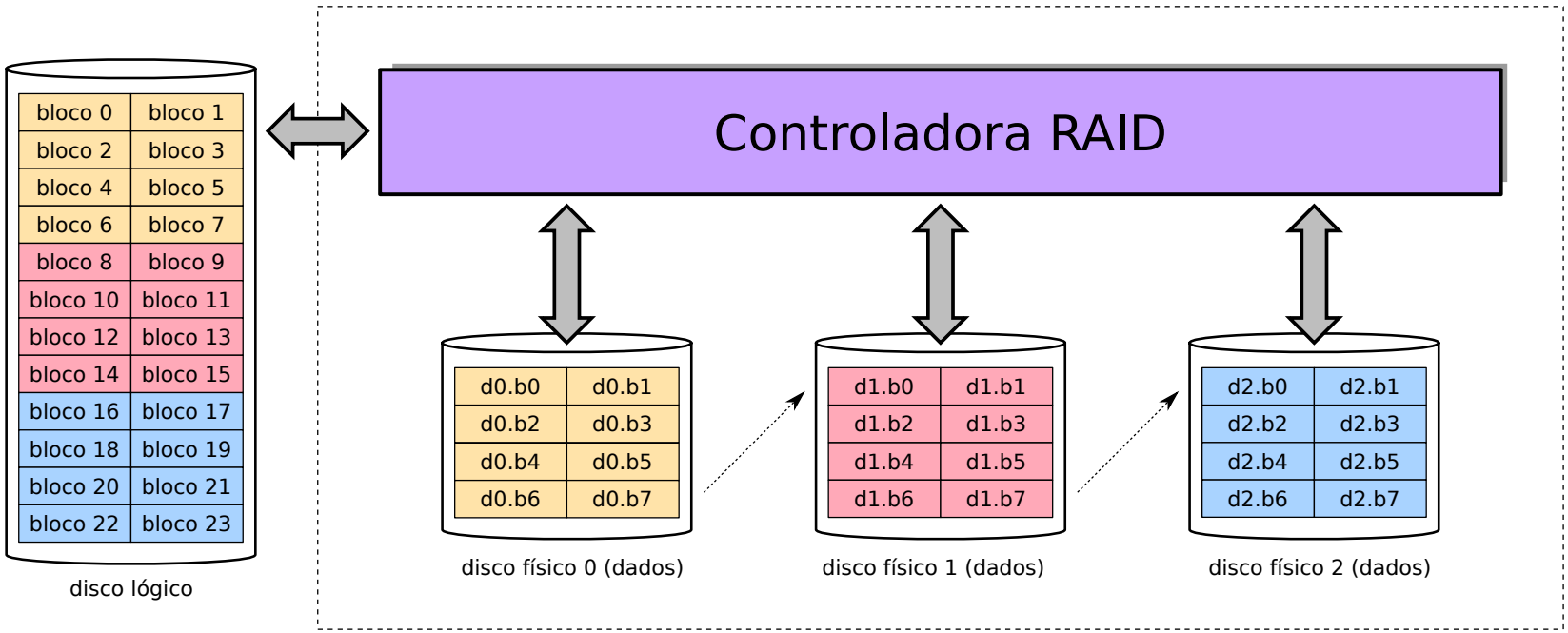
- Criar um **disco lógico** a partir de discos físicos
- **Operações em paralelo** permitem maior desempenho
- **Redundância** (cópias) permitem tolerar falhas
- Implementado em hardware dedicado ou software
- Opera com blocos (abaixo dos arquivos)

Níveis RAID

- **RAID 0** - soma de discos (linear ou *stripping*)
- **RAID 1** - espelhamento de discos
- RAID 2 - redundância de bits (não usado)
- RAID 3 - redundância de bytes (não usado)
- RAID 4 - redundância de blocos, disco de paridade
- **RAID 5** - redundância de blocos, blocos de paridade distribuídos
- **RAID 6** - dois blocos de paridade, para tolerar mais erros
- **RAID 1+0 ou 0+1** - combinações de RAID 0 e 1
- ...

RAID 0 linear

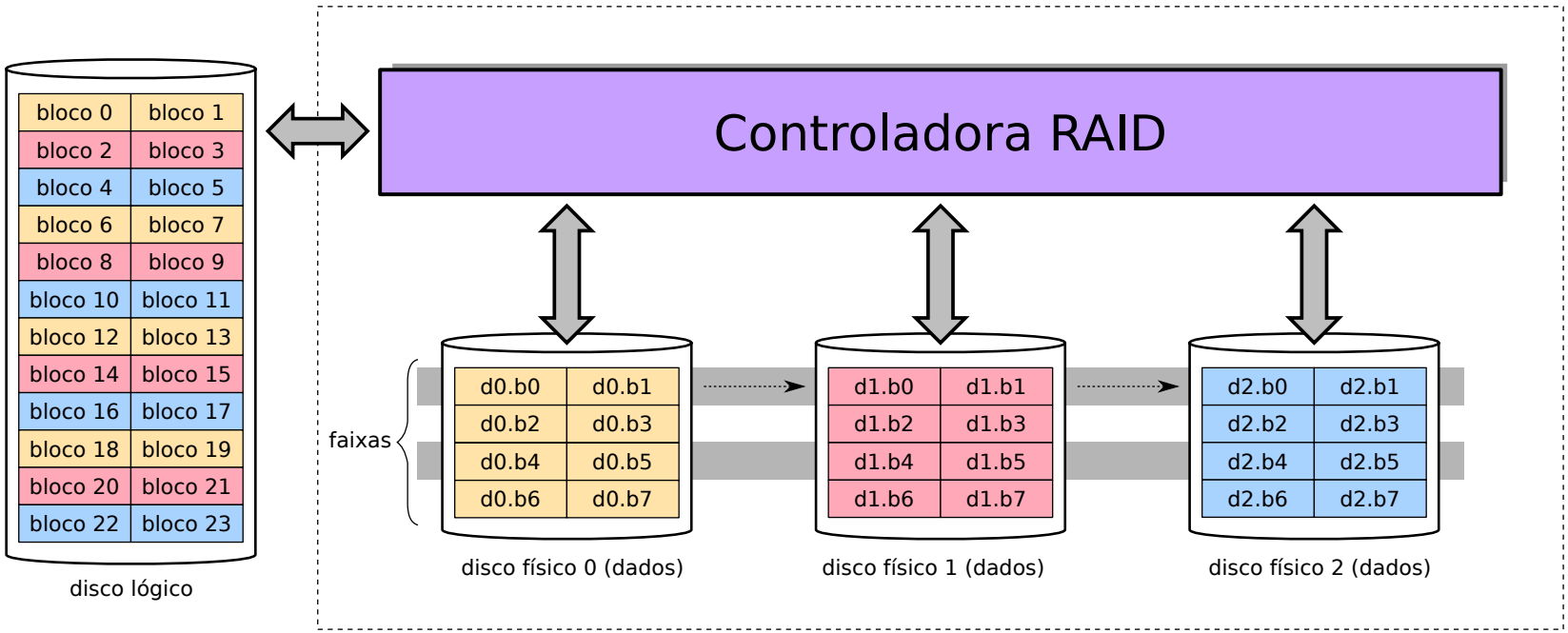
Estratégia: concatena discos em sequência



Mais espaço e velocidade, sem redundância.

RAID 0 *stripping*

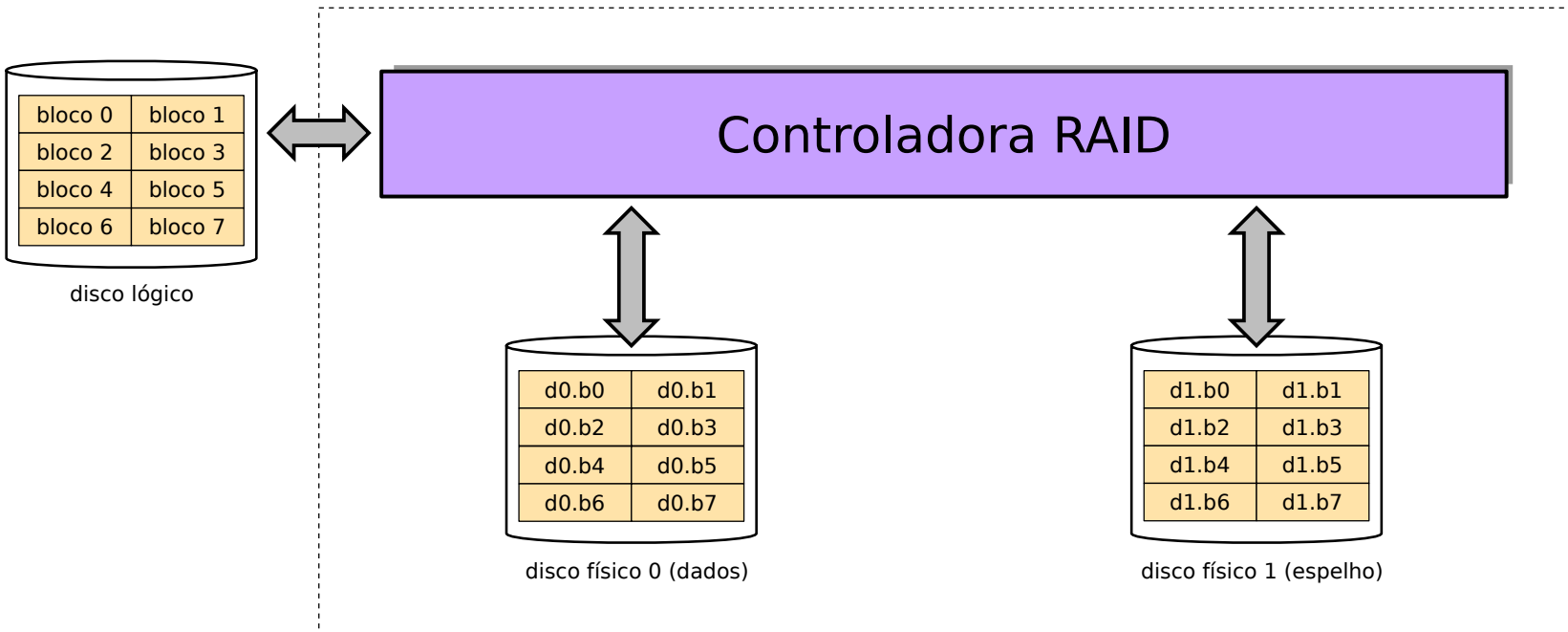
Estratégia: concatena discos em faixas de blocos



Desempenho mais equilibrado entre discos.

RAID 1

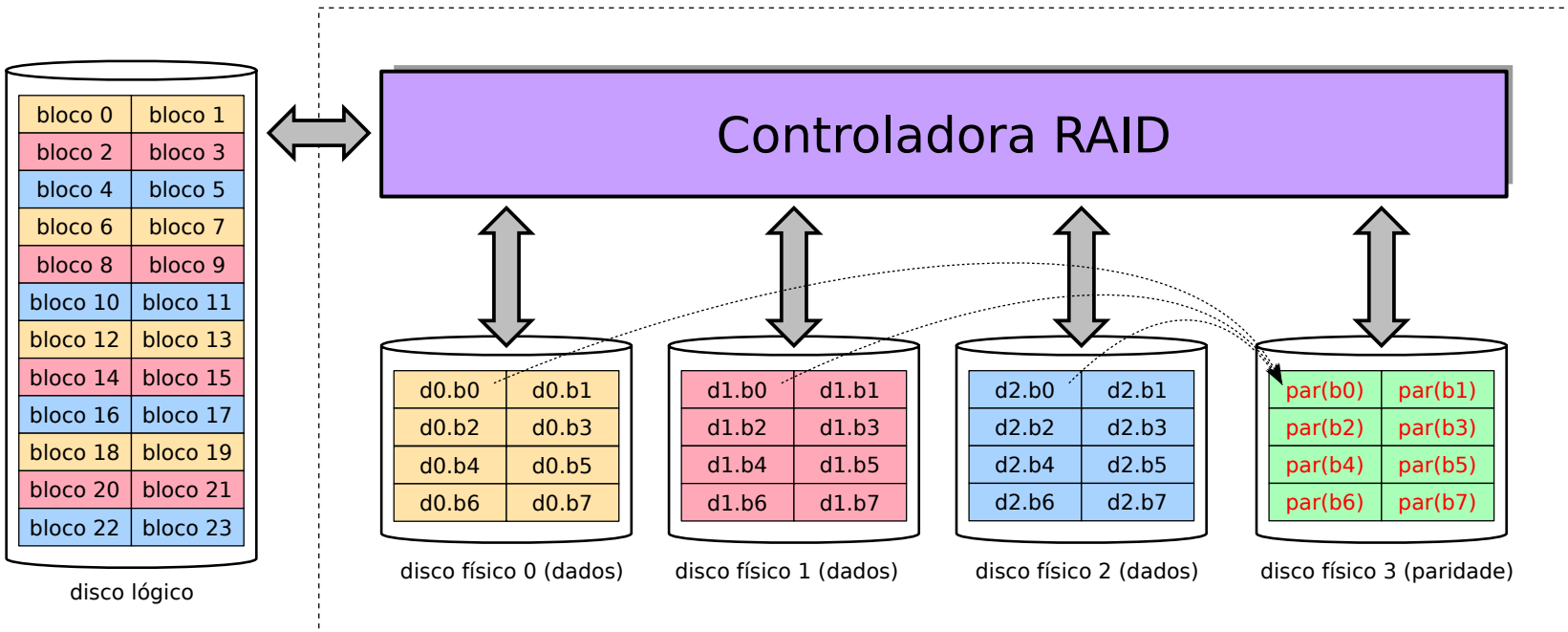
Estratégia: espelhamento (cópias do disco)



Boa velocidade, tolera falhas de disco, mas tem alto custo.

RAID 4

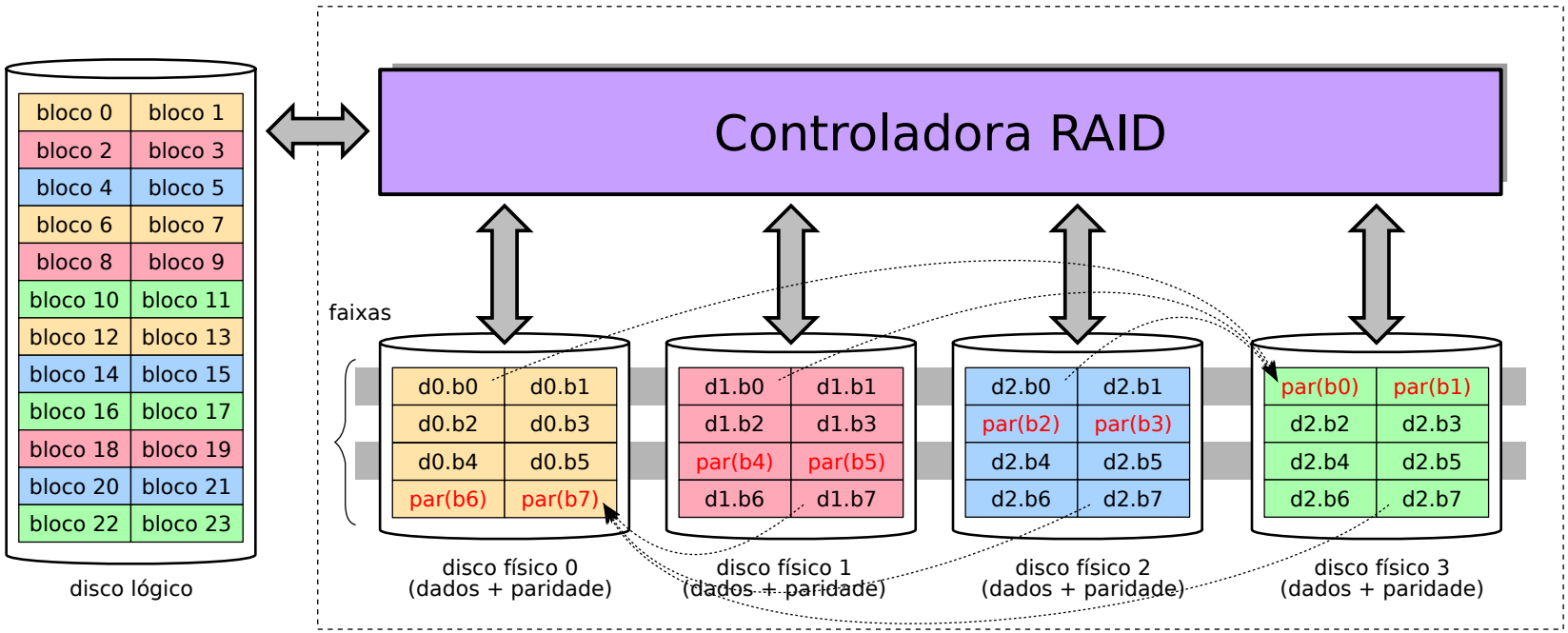
Estratégia: disco com blocos de paridade dos demais discos



Não é usado, base conceitual do RAID 5.

RAID 5

Estratégia: blocos de paridade espalhados nos discos



Mais velocidade com tolerância a falhas e baixo custo.

Comparação de níveis RAID

Considerando arranjos com N discos iguais de tamanho T, velocidade de leitura L e de escrita E:

Estratégia	Vel. leit	Vel. escr	Espaço	Falhas	Discos
RAID 0 linear	$L \times N$	$E \times N$	$T \times N$	0	≥ 2
RAID 0 strip	$L \times N$	$E \times N$	$T \times N$	0	≥ 2
RAID 1	$L \times N$	E	T	$N-1$	≥ 2
RAID 4	$L \times (N-1)$	E	$T \times (N-1)$	1	≥ 3
RAID 5	$L \times N$	$E \times N/4$	$T \times (N-1)$	1	≥ 3
RAID 6	$L \times N$	$E \times N/6$	$T \times (N-2)$	2	≥ 4

Dispositivos de estado sólido

Armazenam dados usando semicondutores, sem partes mecânicas

Vantagens:

- tamanho físico
- rapidez
- silêncio
- resistência a choques
- consumo energético

Desvantagens:

- custo por MB
- capacidade
- complexidade

Exemplos: drives SSD, pendrives, cartões SD

Memórias flash

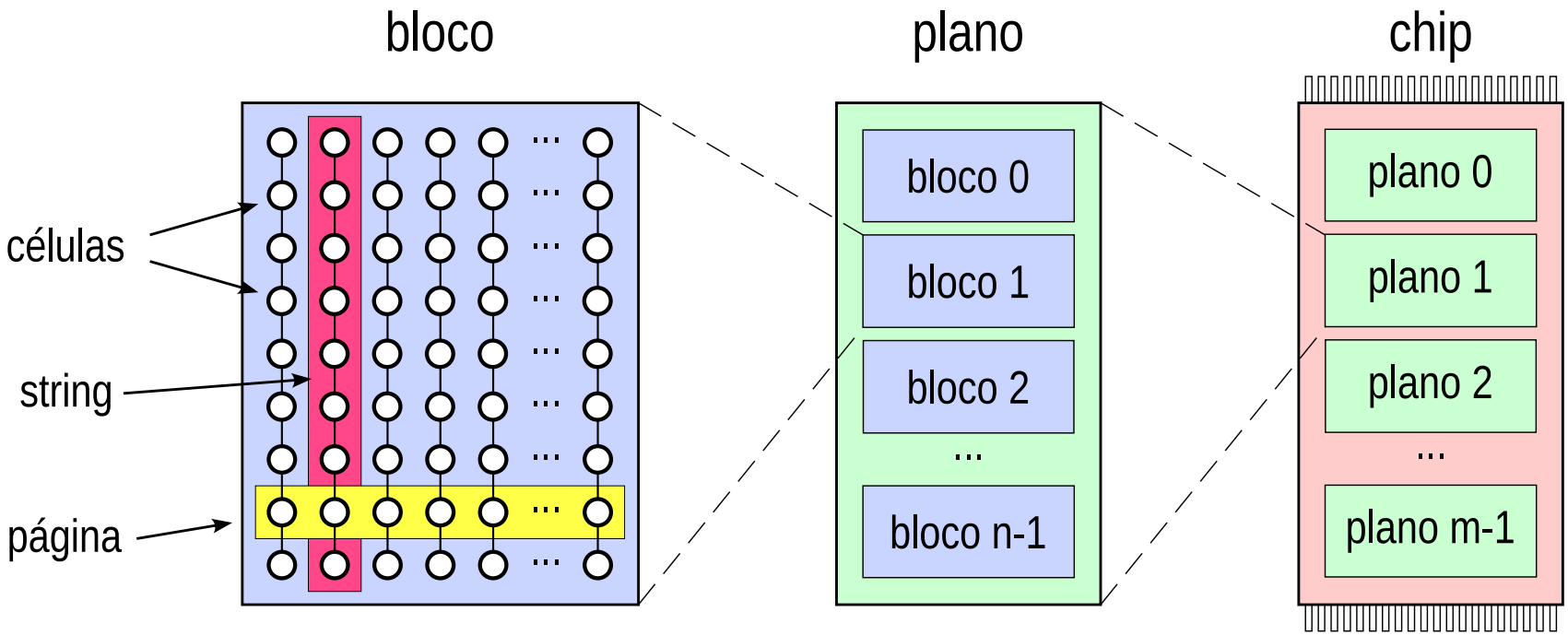
Célula de memória flash:

- circuito que armazena valor lógico sem estar energizado
- SLC - Single Level Cell: armazena 1 bit (0 ou 1)
- MLC - Multi-Level Cell: armazena 2, 3 ou 4 bits
- Podem ser de tipo NOR ou NAND

Estrutura de um chip Flash:

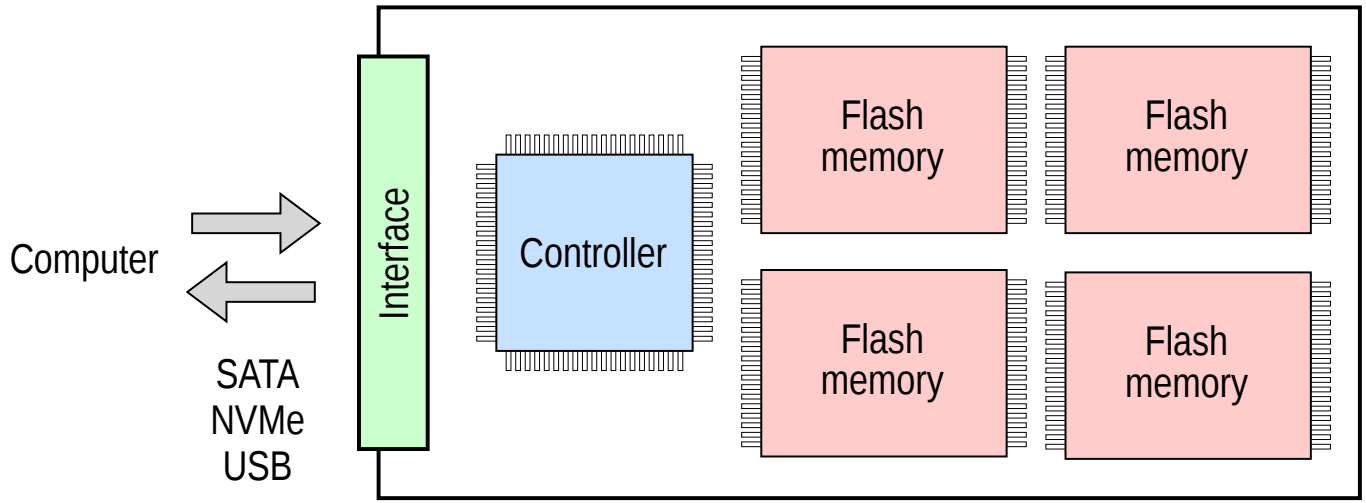
- **String**: grupo de células flash ligadas em série
- **Bloco**: grupo de strings em paralelo
- **Página**: conjunto de células alinhadas em um bloco
- **Plano**: grupo de blocos
- **Chip**: grupo de planos em um invólucro

Memórias Flash



Exemplo: página com 4.096 bytes, bloco com 64 páginas, plano com 2.048 blocos, chip com 8 planos = 4 GBytes (valores típicos)

Estrutura de um SSD



Funções do controlador:

- Interface c/ computador
- Tradução de endereços
- Escrita e apagamento
- Nivelamento de desgaste
- Buffer/cache
- Correção de erros
- Gestão de defeitos
- ...

Camada de tradução Flash

FTL - *Flash Translation Layer*

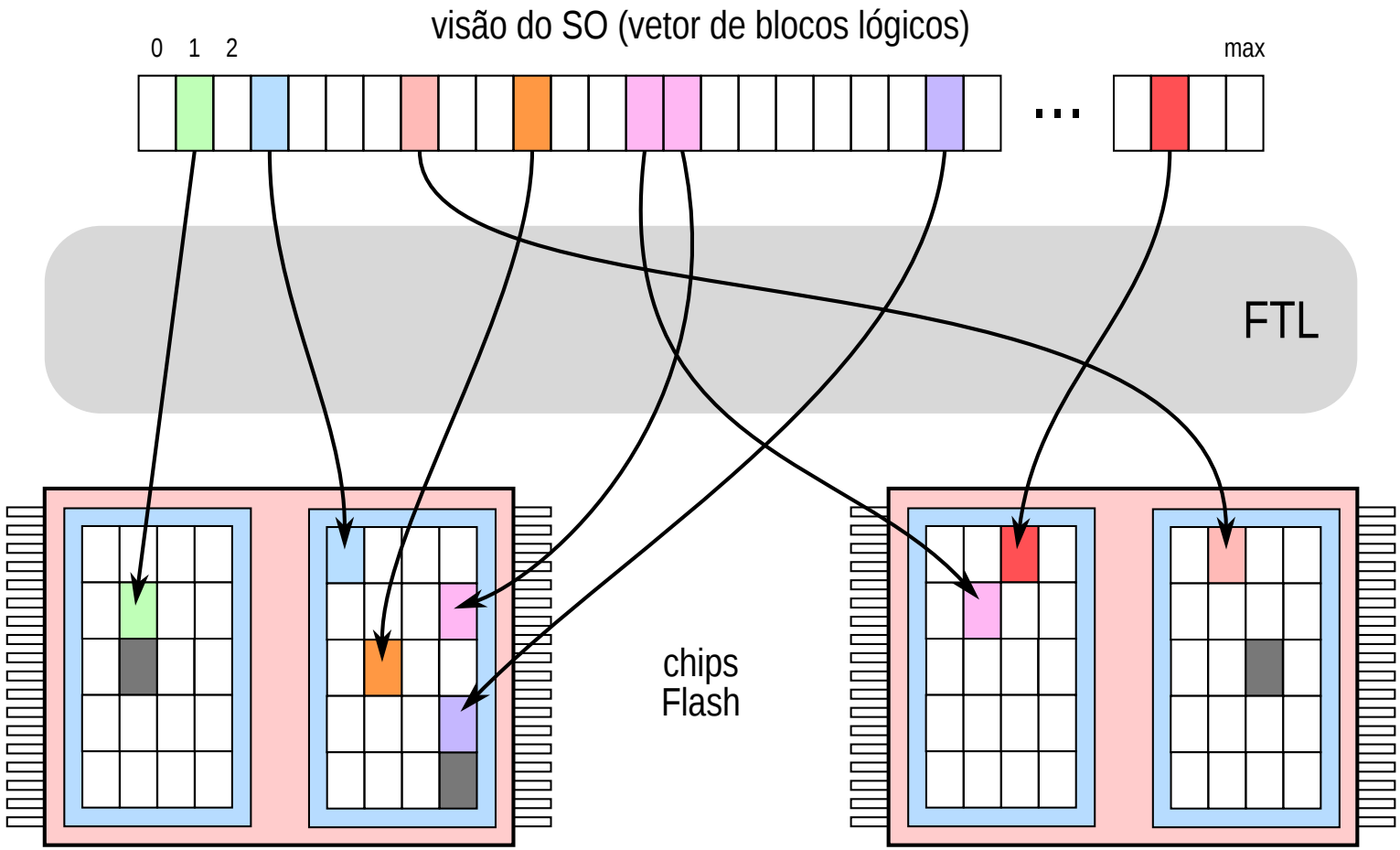
Firmware proprietário, depende do fabricante

Funções da FTL:

- Oferecer ao SO uma visão abstrata do SSD
- Gerenciar escritas e apagamentos
- Gerenciar desgaste das células
- Gerenciar erros e defeitos
- Fazer caching

Mostra o SSD ao SO como um vetor de blocos numerados

Camada de tradução Flash



Escrita e apagamento

Restrições na operação da memória Flash:

- Dados devem ser lidos e escritos em páginas
- Páginas podem ser lidas sem restrições
- Cada página deve ser apagada antes de ser escrita
- Páginas só podem ser apagadas em blocos
- Células flash têm um número limitado de apagamentos

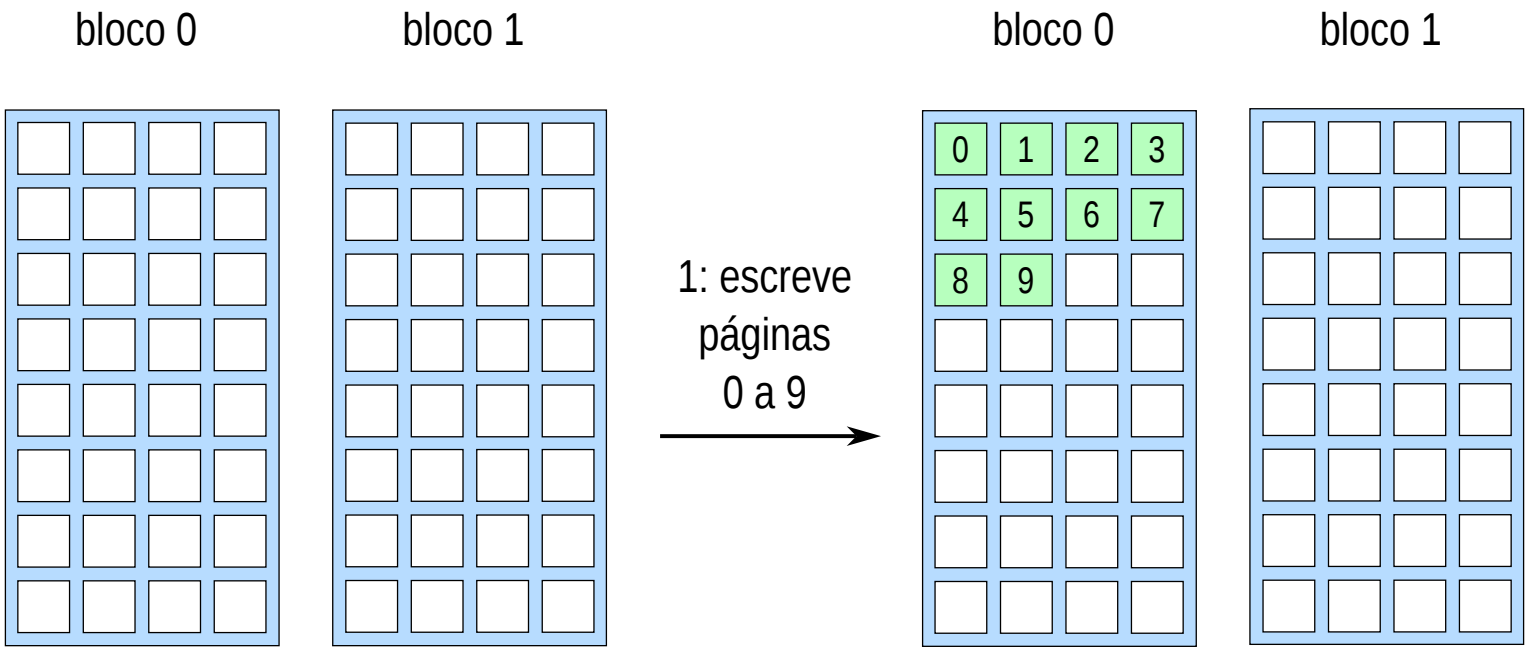
Limite: 5.000 a 100.000 apagamentos (depende do hardware)

Controlador executa **estratégia de escrita e apagamento**

Escrita e apagamento - 1

O SO escreve as páginas lógicas 0 a 9 no SSD

O controlador as aloca no início do bloco 0

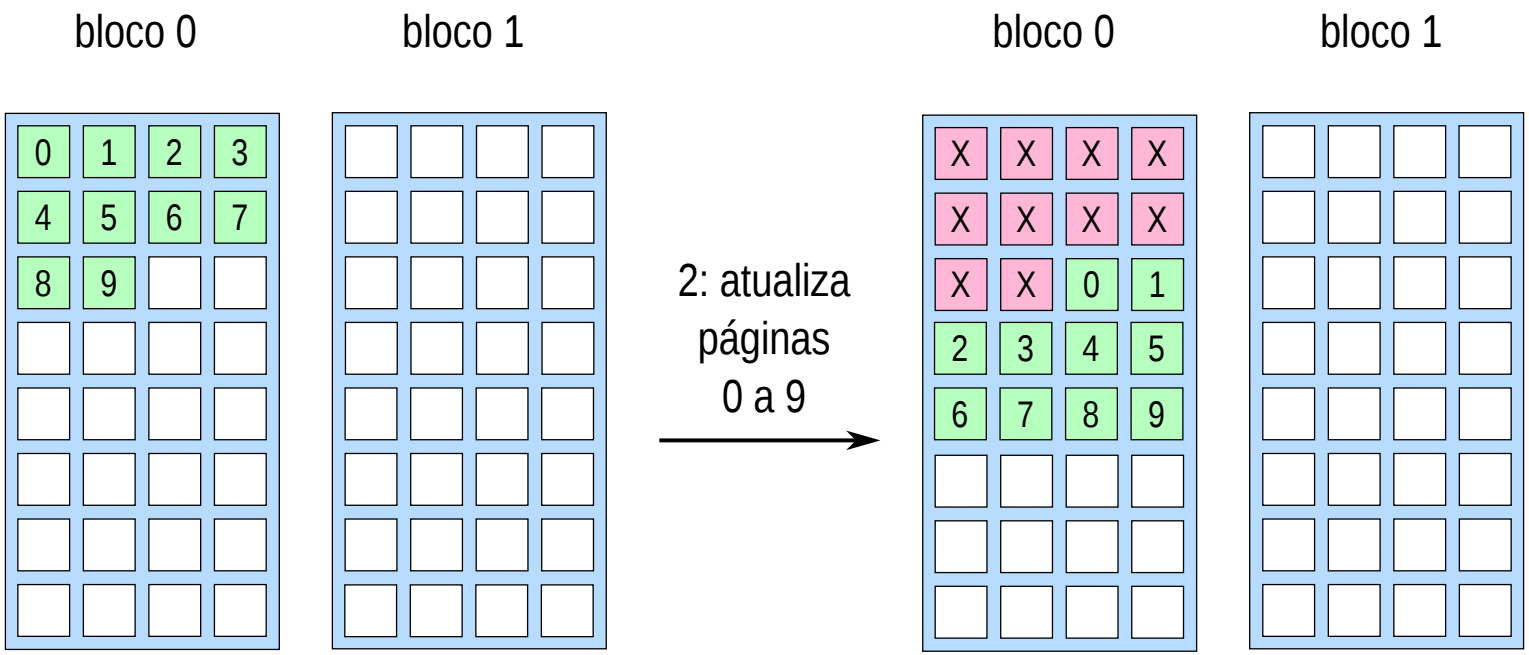


Escrita e apagamento - 2

O SO altera as páginas lógicas 0 a 9

As páginas alteradas são escritas em novas páginas físicas

As páginas anteriores são marcadas como inválidas

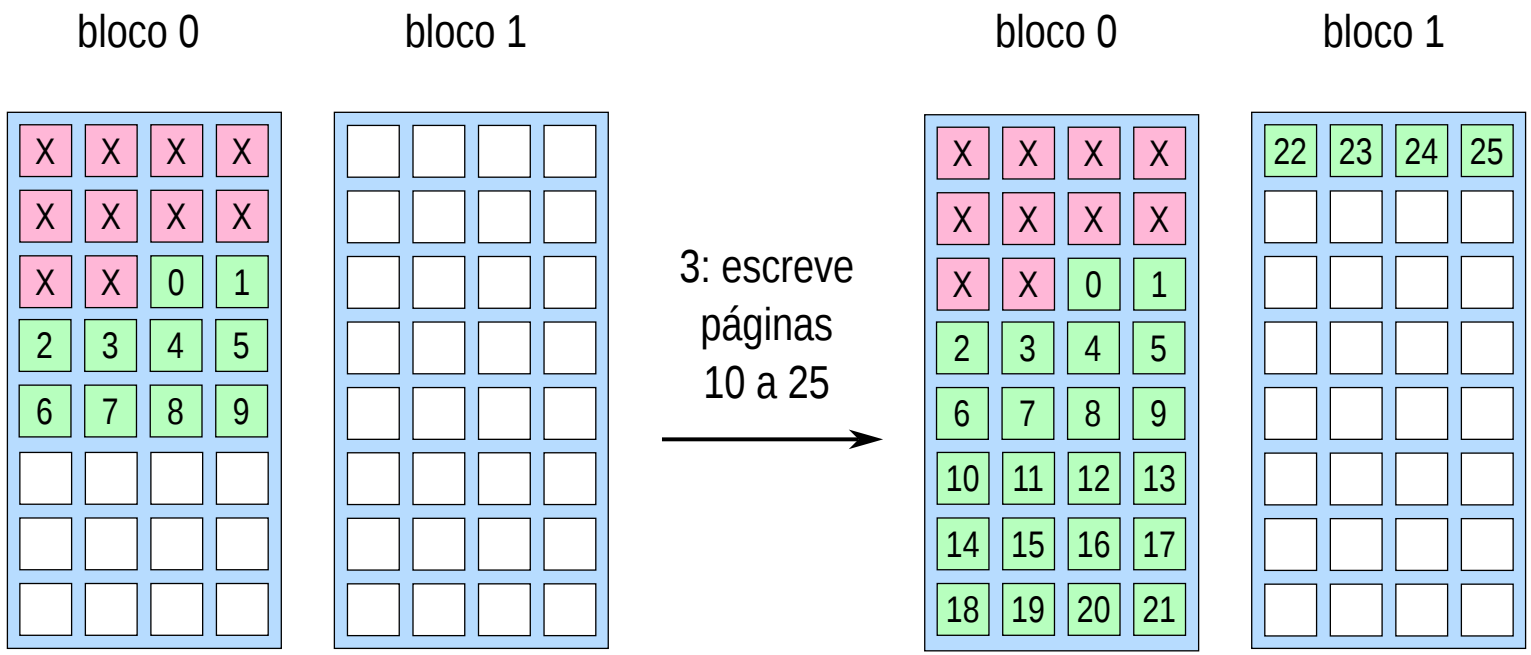


Escrita e apagamento - 3

O SO escreve novas páginas lógicas (10 a 25)

O controlador as aloca em páginas físicas livres

O bloco 0 não pode receber mais escritas até ser apagado

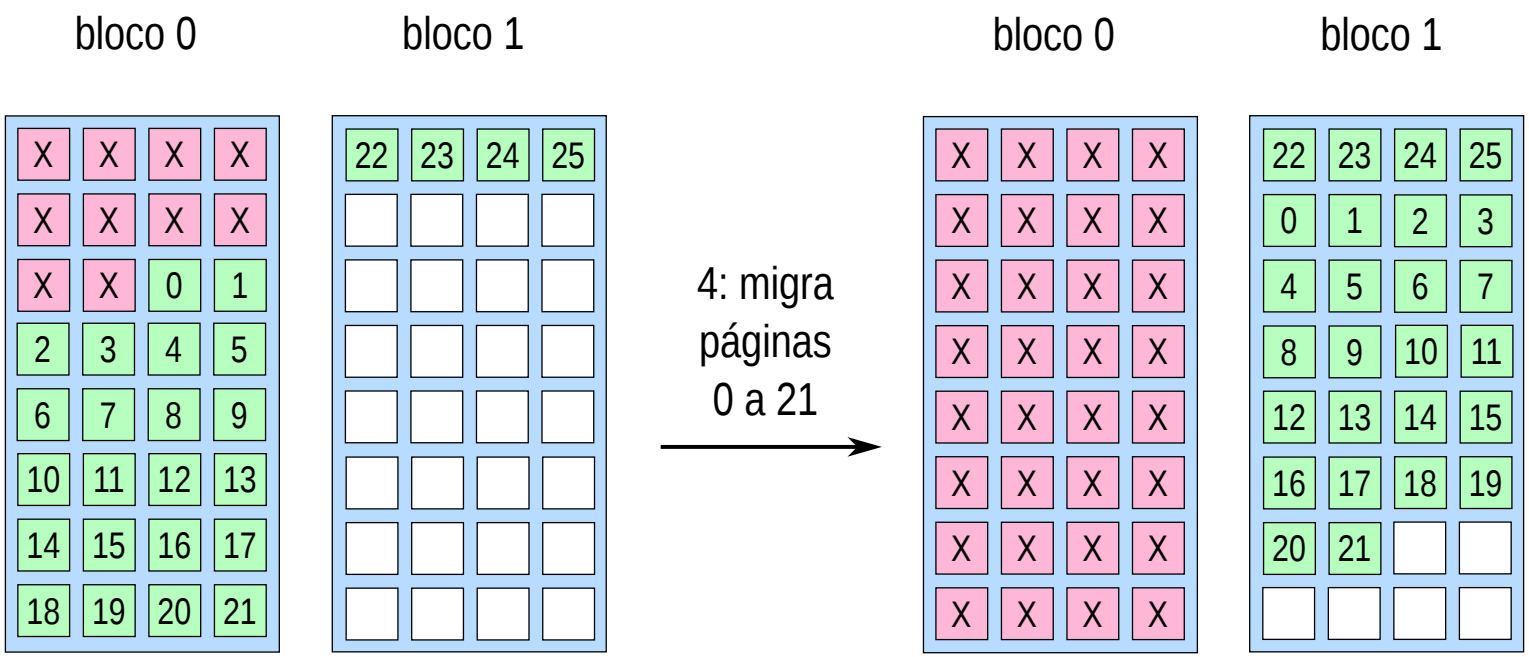


Escrita e apagamento - 4

A “coleta de lixo” visa resgatar blocos de páginas inválidas

As páginas válidas do bloco 0 são copiadas para o bloco 1

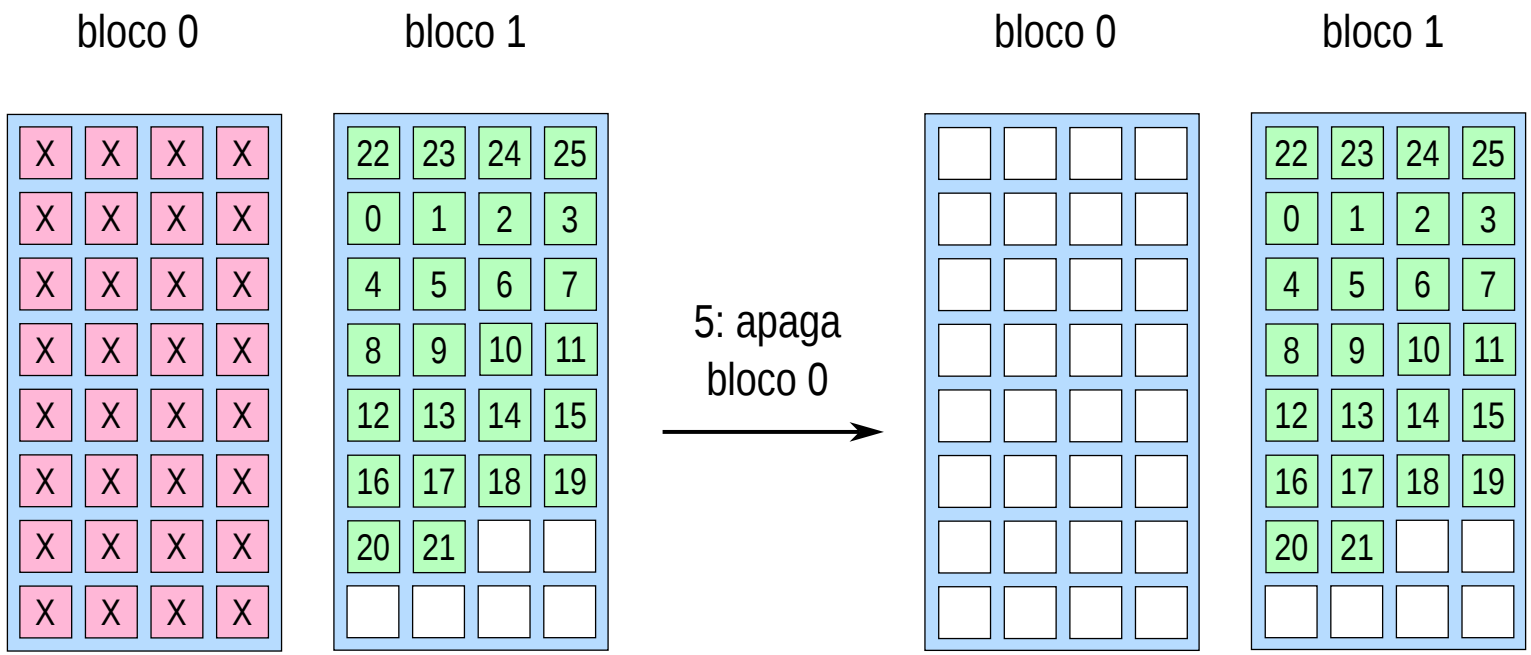
As páginas anteriores são marcadas como inválidas



Escrita e apagamento - 5

O bloco 0 só tem páginas inválidas e pode ser apagado

As páginas do bloco 0 podem ser escritas novamente



Nivelamento de desgaste

Restrições das memórias flash:

- Célula precisa ser apagada antes de ser escrita
- Apagamento é lento e deve ser feito por blocos
- Número de apagamentos é limitado

Wear Leveling:

- Blocos alterados com frequência se desgastam mais cedo
- Blocos pouco alterados têm vida mais longa
- Controlador deve distribuir o desgaste entre os blocos

Estratégia:

- Contar o número de apagamentos de cada bloco
- Escolher páginas em blocos apagados menos vezes

Nivelamento de desgaste

Tipos de nivelamento de desgaste:

- **Dinâmico**: considera só as páginas livres
- **Estático**: migra páginas ocupadas com baixo desgaste

Justificativa:

- Muitos arquivos são pouco alterados (configuração, executáveis, bibliotecas).
- Suas páginas têm menos desgaste que as demais
- Migrá-las permite equilibrar melhor o desgaste geral

Open-channel SSD: não implementa a FTL, o SO faz tudo

O comando TRIM

O SO gerencia arquivos:

- SO cria, altera e apaga arquivos no disco
- Apagar arquivo = remover entrada do diretório
- Blocos de dados do arquivo podem ser reusados

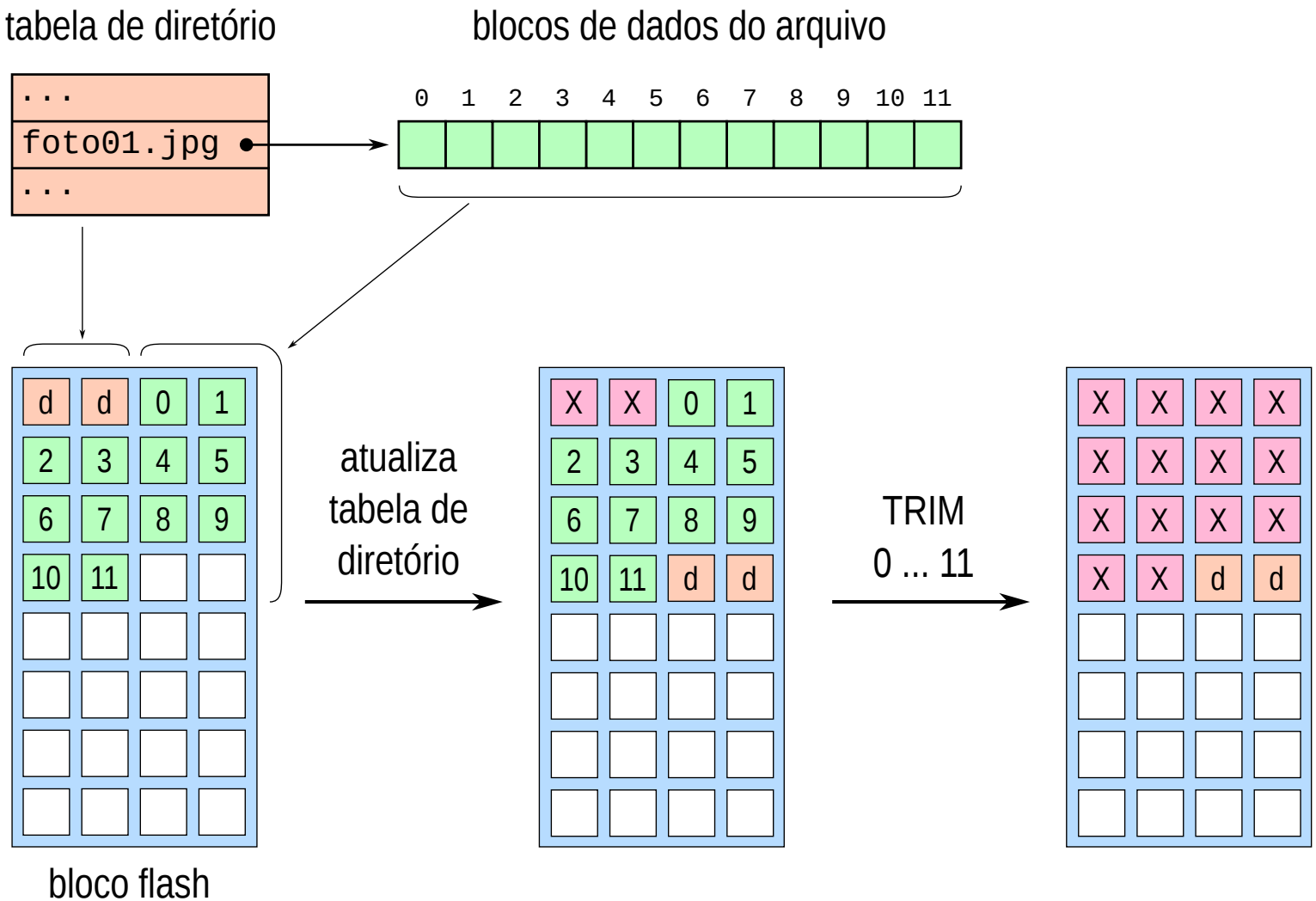
Problema no SSD:

- Controlador não sabe sobre blocos liberados
- Espaço em flash não é efetivamente liberado
- Páginas permanecem válidas até a próxima escrita

Solução: o comando `TRIM` (implementado pelo SSD)

- O SO informa o SSD sobre blocos liberados
- O controlador invalida as páginas correspondentes

O comando TRIM - remoção de um arquivo



Amplificação de escritas

Ao escrever uma página, o controlador:

- pode invalidar uma versão anterior da página
- pode precisar fazer coleta de lixo
- precisa atualizar as tabelas da FTL

Uma escrita do SO $\rightsquigarrow$ várias escritas na memória flash

Fator de **Amplificação de Escrita** (WAF):

$$WAF = \frac{\text{bytes escritos na memória flash}}{\text{bytes escritos pelo SO}}$$

Influenciado pela carga de trabalho e preenchimento do disco

Provisionamento em excesso

Espaço livre no SSD é importante:

- Coleta de lixo precisa de espaço livre
- Nivelamento de desgaste precisa de espaço livre
- SSDs saturados precisam de coletas mais frequentes
- Desempenho e vida útil são prejudicados
- A própria FTL precisa de espaço para suas tabelas
- Células gastas/defeituosas precisam ser substituídas

Provisionamento em excesso:

- Dar ao controlador espaço adicional em flash
- Esse espaço adicional não é visível ao SO

Provisionamento em excesso

Varia entre 0% e 28% (maior em SSDs de maior qualidade):

Capacidade física real	**Capacidade anunciada**	**Prov. em excesso**	**Aplicação**
128 GB	128 GB	0%	SSD básico
128 GB	120 GB	7%	leitura intensiva
128 GB	100 GB	28%	escrita intensiva

Mais espaço livre: diferença entre capacidade em GB (1024^3 bytes) e a capacidade em GiB (1000^3 bytes):

$\rightarrow$ Da ordem de 7,4% para um SSD de 500 GB.

Interface de acesso

Padrões de interface do controlador:

Padrão	velocidade	protocolo	aplicação	status
IDE	1 Gbit/s	paralelo	desktops	obsoleto
SCSI	2,5 Gbit/s	paralelo	servidores	obsoleto
SATA	6 Gbit/s	serial	desktops	atual
SAS	12 Gbit/s	serial	servidores	atual
NVMe	30 Gbit/s	paralelo	SSDs	atual
USB MSC	5 Gbit/s	serial	mídia móvel	atual