



Organização dos Ficheiros no Disco

Sistemas Operativos – DEI - IST

1



Organização do Disco

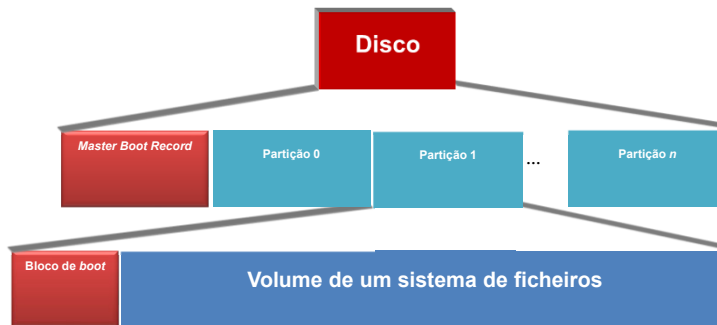
- A gestão do disco é feita em blocos de tamanho fixo com o objetivo de otimizar o acesso ao disco.
- Um bloco é a unidade mínima que pode ser indexada diretamente pelo sistema operativo
- Tamanho dos blocos
 - Quanto maior for o bloco maior é a taxa de transferência bruta pois os tempos de posicionamento e latência de sectores consecutivos são quase nulos, mas a taxa de transferência efetiva depende do número de *bytes* dentro do bloco com informação útil.
 - Se um bloco estiver apenas meio cheio a sua transferência efetiva é metade da transferência bruta, o que significa que quanto maiores forem os blocos, maiores serão as diferenças entre as taxas de transferência bruta e efetiva

Sistemas Operativos – DEI - IST

2



Visão de um Dispositivo do Tipo Disco



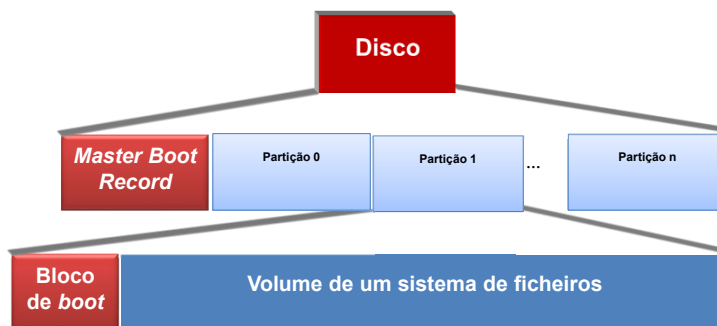
- Uma **partição** é uma subdivisão lógica de um dispositivo físico (disco, pen, etc.), dividida em blocos de tamanho fixo.
- Uma partição é vista pelo sistema operativo como um vetor de blocos ordenado a partir do número zero
- Cada partição é um contexto independente das outras, não existindo ficheiros repartidos por partições diferentes

Sistemas Operativos – DEI - IST

3



Visão de um Dispositivo do Tipo Disco



- O *Master Boot Record* (MBR) e o bloco de *boot* são entidades fundamentais para localizar e executar um sistema operativo
- O MBR possui código, geralmente independente do sistema operativo, que localiza a partição que contém o sistema operativo a executar e transfere a execução para o código existente no primeiro bloco dessa partição – o bloco de *boot*.

Sistemas Operativos – DEI - IST

5



Boot

- O bloco de **boot** possui um programa específico de cada sistema operativo que sabe ler o sistema de ficheiros onde o sistema operativo se encontra, carregar o sistema operativo e executá-lo.
- Quando só existe uma partição com sistema operativo, o código do MBR escolhe imediatamente essa partição para transferir a execução durante a fase de *boot*
- Quando existem mais partições, usualmente é o utilizador que escolhe a partição a executar. Neste caso o código existente no MBR diz-se que é um *boot loader* porque permite escolher entre vários sectores de *boot*, de várias partições, para executar.

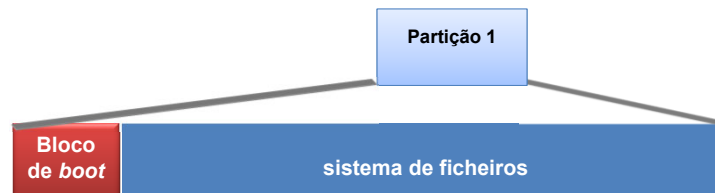
Sistemas Operativos – DEI - IST

6



Organizações dos Ficheiros

- Dada esta organização do disco o problema interessante do ponto de vista de sistema operativo é como organizar os ficheiros
- Vamos ver três alternativas que correspondem a evoluções de sistemas reais



Sistemas Operativos – DEI - IST

7

Organização em Lista

Esta estrutura é muito simples mas tem importantes limitações:

- O tempo necessário para localizar um ficheiro através do seu nome.
 - Os nomes dos ficheiros estão espalhados pelo disco e a sua localização depende da dimensão do ficheiro anterior, é necessário percorrer toda a lista comparando os nomes dos ficheiros um a um;
- A fragmentação do disco (fragmentação externa)
 - Impossível reaproveitar totalmente o espaço previamente ocupado por ficheiros grandes que só podem ser usados por ficheiros de dimensão igual ou inferior,
- Limite ao crescimento dos ficheiros
 - O espaço ocupado por cada ficheiro ser contínuo limita a capacidade de extensão de um ficheiro ao início de outro

Sistemas Operativos – DEI - IST

8

Tabela de Diretórios

- Resolver o problema da pesquisa de nomes
 - Criar uma área no disco para os diretórios onde todos os nomes dos ficheiros estão juntos
 - Aumenta a eficiência da procura de um ficheiro dado o seu nome
- Resolver o problema da fragmentação externa e crescimento de ficheiros
 - A área de dados é composta por blocos e um ficheiro utilizaria os blocos necessários que podem estar dispersos no disco
- Necessário resolver o problema de como indexar os blocos de dados

Sistemas Operativos – DEI - IST

10



Na Origem dos Computadores Pessoais

- O sistema de ficheiros do CP/M (um dos primeiros para PC, 1977) utilizava uma estrutura deste tipo



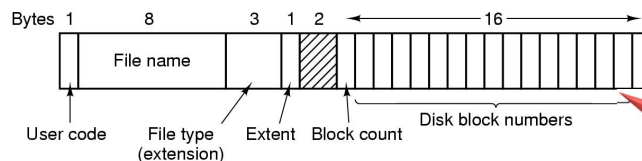
Sistemas Operativos – DEI - IST

11



Sistema de Ficheiros do CP/M

- Estrutura de uma entrada do diretório do sistema de ficheiros do CP/M:



- Neste sistema:
 - cada bloco possuía 1 Kbyte (por omissão)
 - mapa de blocos com 16 entradas, dimensão máxima de um ficheiro = 16 KBytes
- Como aumentar a dimensão máxima dos ficheiros?
 - Aumentar o mapa de blocos → ineficiente para fich. pequenos
 - Aumentar o tamanho dos blocos → maior fragmentação

Sistemas Operativos – DEI - IST

12



Sistema de Ficheiros do MS-DOS

- Possui uma estrutura de sistema de ficheiros semelhante ao CP/M
- Em vez de um mapa de blocos por ficheiro, no MS-DOS existe:
 - uma tabela de blocos global partilhada por todos os ficheiros
 - esta tabela única é tão representativa da estrutura do sistema de ficheiros que deu origem ao nome do sistema de ficheiros: **o sistema de ficheiros FAT** (*File Allocation Table*).

History

DOS and Timeline of DOS operating systems
MS-DOS was a renamed form of 86-DOS – owned by Seattle Computer Products, written by Tim Paterson. Development of **86-DOS took only six weeks**, as it was basically a clone of Digital Research's CP/M (for 8080/Z80 processors), ported to run on 8086 processors and with two notable differences compared to CP/M: an improved disk sector buffering logic, and the introduction of FAT12 instead of the CP/M filesystem

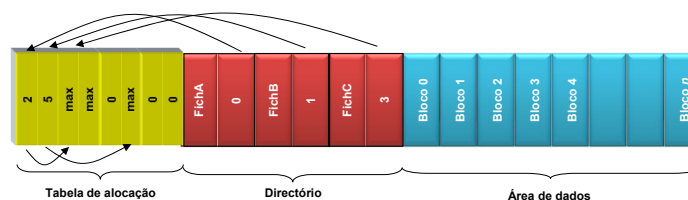


Sistemas Operativos – DEI - IST

13



File Allocation Table (FAT)



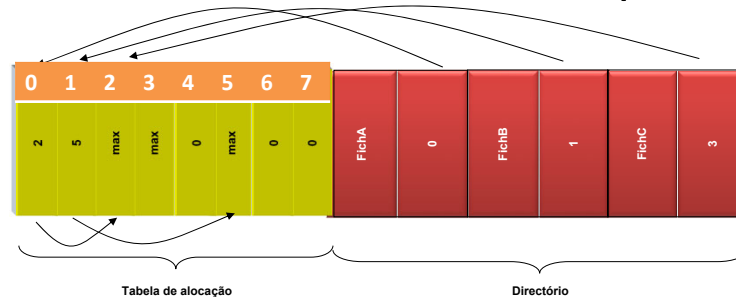
- A partição contém três secções distintas:
 - a tabela de alocação (*File Allocation Table*),
 - Directoria com os nomes dos ficheiros presentes no sistema de ficheiros, e
 - Área de dados dividido em blocos dimensão fixa
- FAT é um vetor composto por 2^n inteiros de n bits
 - Designado de FAT-16 ($n=16$) ou FAT-32 ($n=32$)
- Dimensionado para:
 - Caber em memória RAM (FAT é carregada do disco para a RAM quando o FS é montado)
 - Ter tantas entradas quanto o número de blocos de dados na partição em disco

Sistemas Operativos – DEI - IST

14



Sistemas de Ficheiros do Tipo FAT



- Os blocos de um ficheiro são determinados assim:
 - 1º bloco: indicado por um índice na FAT guardado na respetiva entrada no diretório.
 - restantes blocos: referenciados em lista ligada pelas entradas da FAT
- As entradas da FAT:
 - com o valor zero indicam que o respetivo bloco está livre,
 - com valores diferentes de zero indicam que o respetivo bloco faz parte de um ficheiro
 - Com todo os bits a 1 indica que é o ultimo bloco

Sistemas Operativos – DEI - IST

15



Desvantagens do FAT

- Elevada dimensão da FAT quando os discos têm dimensões muito grandes:
 - Por exemplo, uma partição 1 *Tbyte*
 - Usando FAT-32 e blocos de 4 *Kbytes*...
 - ...a FAT pode ocupar 1 *GByte* ($1\text{TBytes}/4\text{KBytes} \times 4 \text{ bytes}$)
- Uma tabelas desta dimensão não é possível manter em RAM permanentemente:
 - Ler a FAT do disco, prejudica muito o acesso à cadeia de blocos de um ficheiro

Sistemas Operativos – DEI - IST

16



Organização com Descritores Individuais de Ficheiros

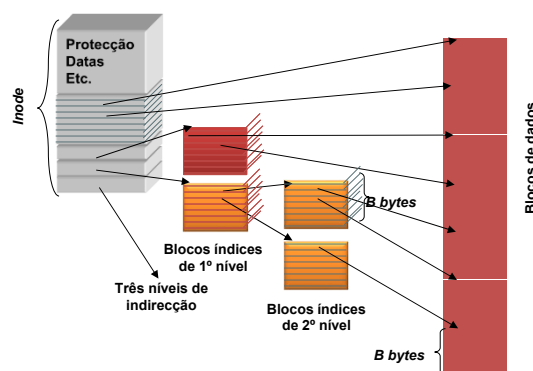
- Criar uma estrutura de indexação em árvore entre as entradas dos diretórios e os blocos de dados do ficheiro
- Manter a descrição dos blocos do ficheiro num descritor próprio de cada ficheiro (no Unix o *i-node*)
 - Para privilegiar os ficheiros pequenos o descritor pode referenciar diretamente os primeiros blocos (ex.: 10 entradas)
 - A 11ª entrada referencia um bloco de dados normal mas que é usado como um bloco de índice (numero de blocos indexáveis obtém-se dividindo o tamanho do bloco pelo tamanho da referência)
 - A 12ª entrada pode ser uma árvore em que o bloco de índices já não referencia blocos de dados, mas outros blocos de índices estabelecendo uma hierarquia de índices (neste caso o numero de bloco indexáveis é o quadrado do valor anterior)
 - A 13ª ...

Sistemas Operativos – DEI - IST

17



Referenciação dos Blocos do Ficheiro em Unix



Com blocos de 1 Kbyte e referências de 4 byte, a dimensão máxima de um ficheiro é ~16 Gbyte

$$\text{Dimensão máxima de um ficheiro} = B \times (12 + B/R + (B/R)^2 + (B/R)^3)$$

B é a dimensão em bytes de um bloco de dados

R é a dimensão em bytes de uma referência para um bloco

Sistemas Operativos – DEI - IST

18



i-node (*index node*)

- O *inode* que contem a *metadata* do ficheiro terá este mapa de blocos
- Vetor de 15 entrada
 - 12 diretas para os blocos
 - 13 – endereço do bloco que tem um nível de indexação
 - 14 – endereço do bloco que tem 2 níveis de indexação
 - 15 – endereço do bloco que tem 3 níveis de indexação

Campo	Descrição
i_mode	Tipo de ficheiro e direitos de acesso
i_uid	Identificador do utilizador
i_size	Dimensão do ficheiro
i_atime	Tempo do último acesso
i_ctime	Tempo da última alteração do inode
i_mtime	Tempo da última alteração do ficheiro
i_dtime	Tempo da remoção do ficheiro
i_gid	Identificador do grupo do utilizador
i_links_count	Contador de hard links
i_blocks	Número de blocos ocupado pelo ficheiro
i_flags	Flags várias do ficheiro
i_block[15]	Vector de 15 unidades para blocos de dados
	Outros campos ainda não utilizados

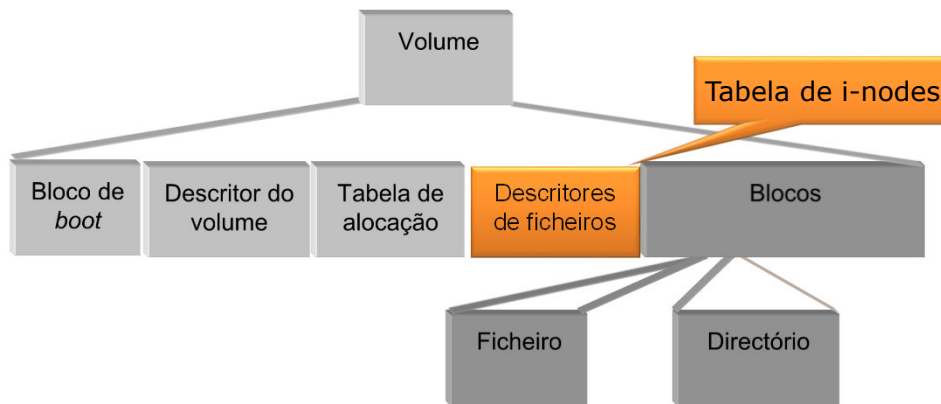
Sistemas Operativos – DEI - IST

19



Tabela de i-nodes no volume

- Mantidos numa tabela em zona própria no volume
- Dentro de um volume, cada *i-node* é identificado por um *i-number* índice do *i-node* na tabela de *i-nodes* no volume



Sistemas Operativos – DEI - IST

20



Bitmap de *inodes* e blocos

- O número máximo de ficheiros, correspondente à dimensão máxima da tabela de *inodes*, que armazena os *inodes* de todos os ficheiros da partição
- Dentro de uma partição, um *inode* é univocamente identificado pelo índice dentro da tabela de *inodes*
- Em cada partição duas outras tabelas, o *bitmap* de *inodes* e o *bitmap* de blocos indicam respetivamente quais as posições que estão livres e ocupadas nessas estruturas.

Sistemas Operativos – DEI - IST

21



Conclusões

- A organização do disco é em blocos de tamanho fixo.
- Diversas zonas do disco são atribuídas a funções específicas, bloco de *boot*, partições, diretórios, blocos de dados
- A indexação dos blocos de dados de um ficheiro é uma importante parte da organização porque tem elevado impacto no desempenho e na fragmentação do disco
- No Unix a estrutura de indexação é mantida nos *inodes* procurando um equilíbrio entre acesso rápido a ficheiros pequenos e possibilidade de ter ficheiros muito grandes

Sistemas Operativos – DEI - IST

22