# Tópicos em Recuperação de Informação[1]

Nivio Ziviani

[1]Conjunto de transparências elaborado por Nivio Ziviani e Patrícia Correia

# Como Armazenar e Acessar o Vocabulário

O vocabulário de um arquivo invertido armazena os termos que serão usados para buscar na coleção e informações auxiliares para permitir que consultas sejam processadas.

| **Método** | **Armazenamento** |
|---|---|
| Fixed Length Strings | 28 Mbytes |
| Terminated Strings | 20 Mbytes |
| Four-entry blocking | 18 Mbytes |
| Front coding | 15,5 Mbytes |
| Hashing perfeito | 13 Mbytes |

Para um vocabulário de 1.000.000 de termos

# Estruturas

## *Fixed Length Strings*

- Para um vocabulário de 1.000.000 de palavras:
  - 20 bytes: palavra (assumindo que palavras ocupam no máximo 20 bytes).
  - 4 bytes: endereço do arquivo invertido.
  - 4 bytes: $f_t$ (número de docs que contêm o termo $t$).
  - Vocabulário requer 28 Megabytes.

- Esta alternativa é cara. Uma redução pode ser conseguida se todas as palavras forem concatenadas como uma palavra longa.

# Estruturas

## *Fixed Length Strings*

| Termo $t$ | $f_t$ | Endereco em disco |
|---|---|---|
| jezebel | 20 | |
| jezer | 3 | |
| jezerit | 1 | |
| jeziah | 1 | |
| jeziel | 1 | |
| jezliah | 1 | |
| jezoar | 1 | |
| jezrahiah | 1 | |
| jezreel | 39 | |

# Estruturas

## *Terminated Strings*

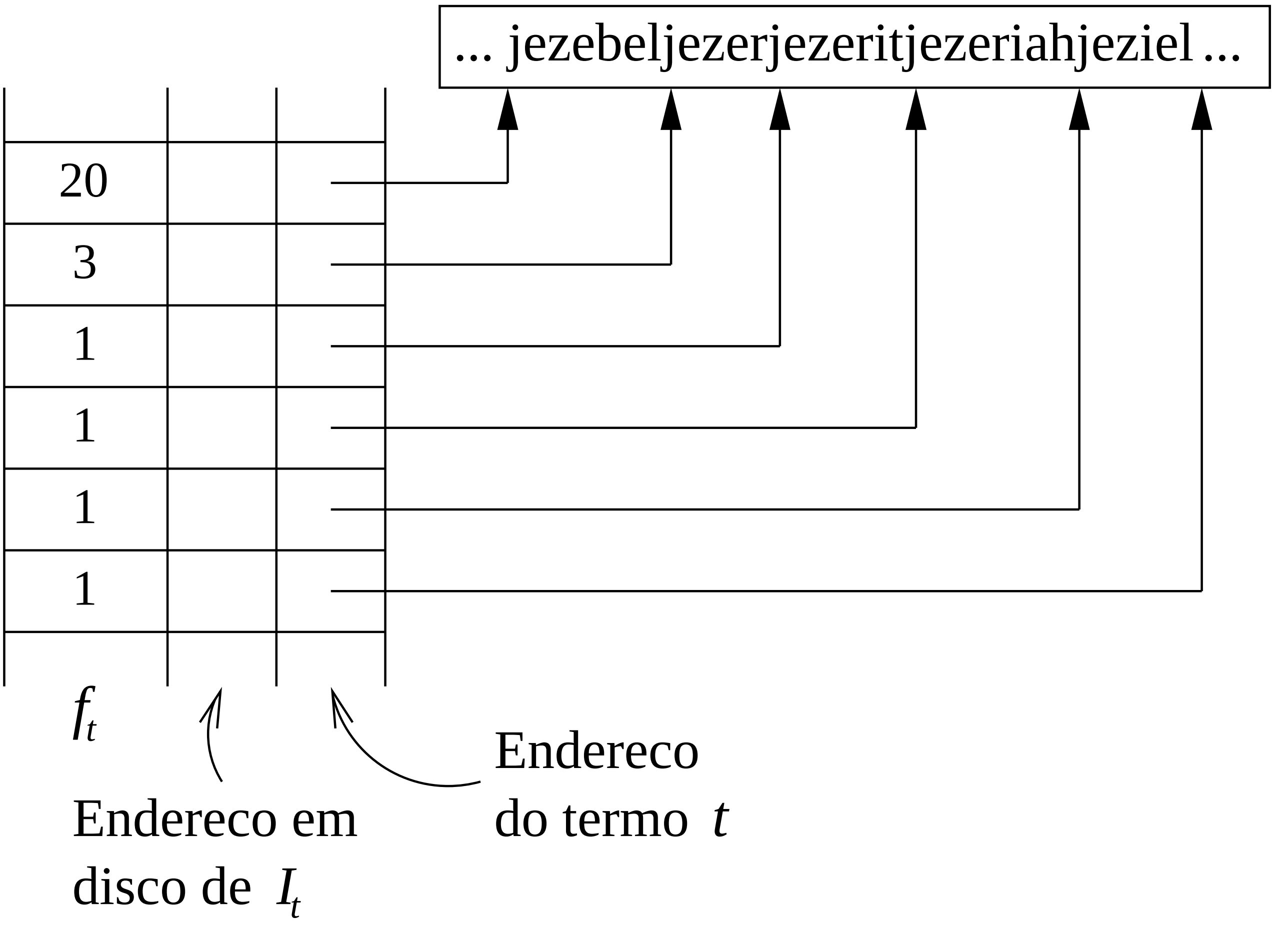
- Cada termo ocupa exatamente o seu tamanho.

- Para um vocabulário de 1.000.000 de palavras:
  - O vocabulário requer 20 Megabytes.

- O espaço requerido pode ainda ser reduzido eliminando muitos dos ponteiros de strings.

# Estruturas

## *Terminated Strings*



# Estruturas

## *Four-entry Blocking*

- 1 byte antes de cada palavra para indicar o seu tamanho.

$$\Downarrow$$

- Para cada grupo de 4 palavras, 12 bytes são ganhos.

$$\Downarrow$$

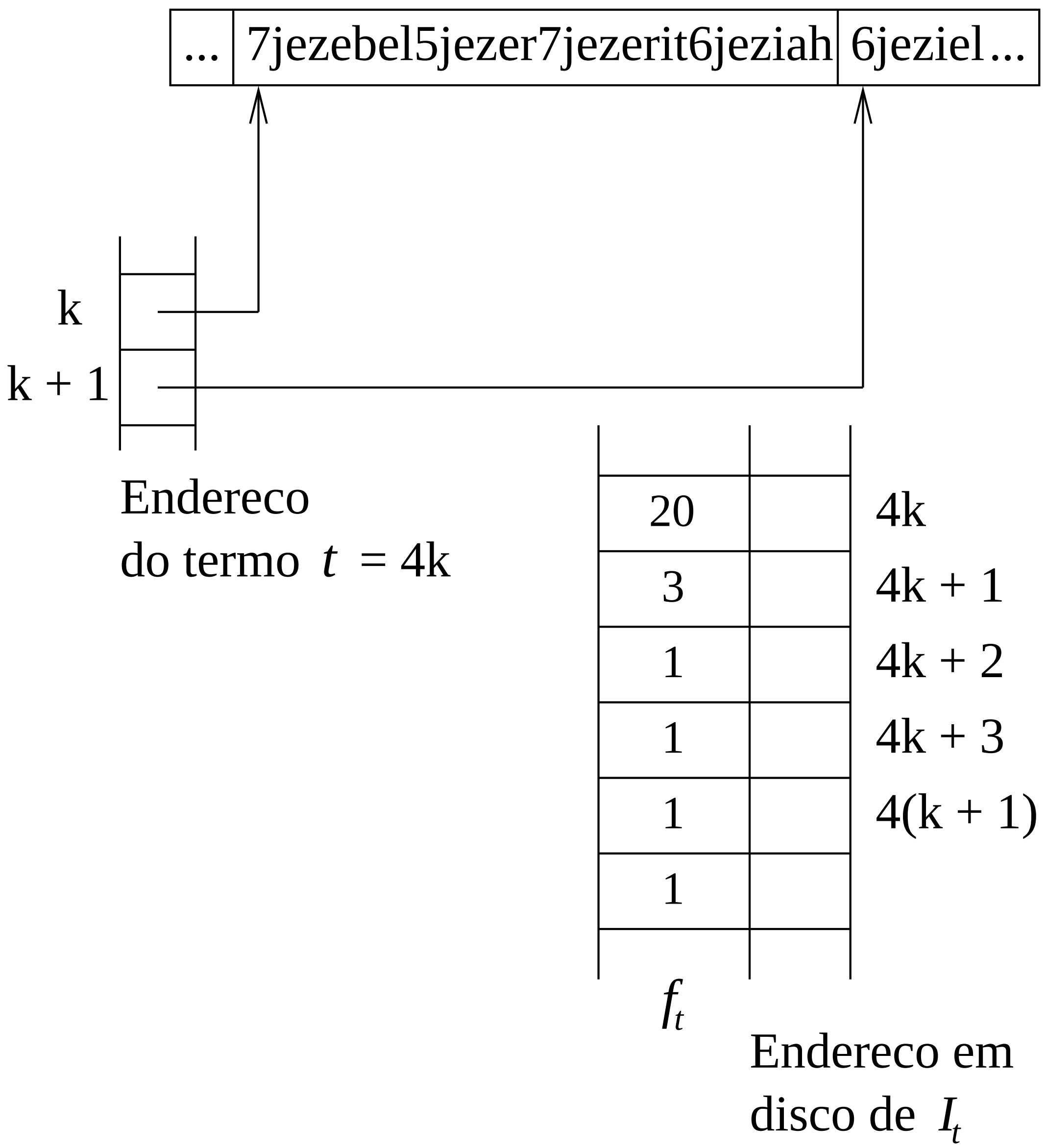
- Ganho de 2 Megabytes. O espaço total necessário é de 18 Megabytes.

- Para blocos de 8 palavras, o ganho é de + 0.5 Megabytes.

- Para blocos de 16 palavras, o ganho é de + 0.25 Megabytes.

# Estruturas

## *Four-entry Blocking*



# Estruturas

## *Front Coding*

- O ganho depende do vocabulário. Na média, 3 a 5 caracteres casam com custo adicional de 1 byte.

$$\Downarrow$$

- Ganho médio é de 2.5 bytes por palavra.

- Regra do dedão:
  - *Front coding* salva aproximadamente 40% do espaço para vocabulário da lingua inglesa.

- Uma estratégia 3-em-4 *front-coding*: salva 4 bytes em cada 3 palavras com custo extra de 2 bytes.

$$\Downarrow$$

- Total de espaço necessário: 15.5 Megabytes.

# Estruturas

## *Front Coding*

| **Palavra** | **Front coding completo** | **Front coding "3-in-4" parcial** |
|---|---|---|
| 7, jezebel | 3, 4, ebel | , 7, jezebel |
| 5, jezer | 4, 1, r | 4, 1, r |
| 7, jezerit | 5, 2, it | 5, 2, it |
| 6, jeziah | 3, 3, iah | 3, , iah |
| 6, jeziel | 4, 2, el | , 6, jeziel |
| 7, jezliah | 3, 4, liah | 3, 4, liah |
| 6, jezoar | 3, 3, oar | 3, 3, oar |
| 9, jezrahiah | 3, 6, rahiah | 3, , rahiah |
| 7, jezreel | 4, 3, eel | , 7, jezreel |
| 11, jezreelites | 7, 4, ites | 7, 4, ites |
| 6, jibsam | 1, 5, ibsam | 1, 5, ibsam |
| 7, jidlaph | 2, 5, dlaph | 2, , dlaph |

A palavra antes de *jezebel* era *jezaniah*.

# Estruturas

## *Front Coding*

- Cada entrada do vocabulário pode ser mais reduzida ainda:

$$f_t \Rightarrow \lceil \log N \rceil \text{ bits, no exemplo anterior.}$$

$$\Downarrow$$

$$f_t + d \approx 28 \text{ bits cada.}$$

$$\Downarrow$$

- Ganho de mais 1 Megabyte.

- É possível fazer melhor?

  Resp.: Sim, não armazenando o vocabulário!

# Referências

[WMB99] Witten I., Moffat A., Bell C. Managing Gigabytes Compressing and Indexing Documents and Images. 2nd Ed., 1999, pag. 161-169.