



# Listas invertidas

---

- Lista de chaves primárias de uma certa chave secundária
  - pode crescer
  - ocupa somente o espaço necessário
  - não é alterada ou rearranjada quando um novo registro é inserido no índice de chave secundária
    - uma nova inserção afeta somente o índice e possivelmente a lista associada ao novo registro, e não as demais listas
- Porém
  - na visão conceitual, tem-se um grande número de arquivos pequenos ...

# Listas invertidas: visão implementação

Improved revision of the composer index

Secondary Index file		Label ID List file	
0	BEETHOVEN	3	
1	COREA	2	
2	DVORAK	7	
3	PROKOFIEV	10	
4	RIMSKY-KORSAKOV	6	
5	SPRINGSTEEN	4	
6	SWEET HONEY IN THE R	9	

  

Label ID	RRN	
0	LON2312	-1
1	RCA2626	-1
2	WAR23699	-1
3	ANG3795	8
4	COL38358	-1
5	DG18807	1
6	MER75016	-1
7	COL31809	-1
8	DG139201	5
9	FF245	-1
10	ANG36193	0

*campo de chave secundária*

*campo com RRN da primeira referência da chave primária na lista invertida*

*campo de chave primária*

*campo com RRN da próxima referência da chave primária na lista invertida, ou -1 caso contrário*



# Listas invertidas: Vantagens

---

- Índice secundário
  - alterado quando insere-se um registro com chave inexistente, ou quando altera-se chave já existente
- Remoção, inserção ou alteração de registros já existentes
  - alteração apenas no arquivo da lista invertida
  - modificação do campo de referência do índice se necessário
- Ordenação do arquivo de índice secundário
  - mais rápida: menos registros e registros menores
- Registros de tamanho fixo
  - facilita a adoção de um mecanismo para reaproveitamento de espaço



# Listas invertidas: Desvantagem

---

- Chaves primárias associadas a uma certa chave secundária não estão adjacentes fisicamente no disco
  - pode ser necessário realizar vários *seeks* para recuperar a lista

Ideal: manter o índice e a lista na memória primária



# Índices Seletivos (*índices secundários*)

---

- Motivação
  - dividir *logicamente* o conteúdo do arquivo de dados em categorias
- Característica
  - criação de vários índices, um para cada categoria
  - exemplos
    - índice secundário de músicas clássicas
    - índice secundário de músicas lançadas depois de 1980
- Dependente da aplicação e uso dos dados



# Binding

---

- Ligação entre a chave de um registro do arquivo de índice e o endereço físico do registro correspondente no arquivo de dados
  - ligação feita pelo campo de referência
- Questão importante
  - quando efetuar a ligação (*binding*), especialmente considerando um *índice secundário*

No projeto de um novo sistema de arquivos, é melhor lidar com a questão de *binding* de forma direta e o mais cedo possível

# Tight Binding (*fortemente ligado*)

Beethoven	167
Beethoven	396
Beethoven	256
Beethoven	77
Corea	132
Dvorak	353
Prokofiev	32
Rimsky	300
Springsteen FF245]	211
Sweet Honey In The	442

32	LON   2312   Romeo and Juliet   Prokofiev ...
77	RCA   2626   Quartet in C Sharp Minor ...
132	WAR   23699   Touchstone   Corea ...
167	ANG   3795   Symphony No. 9   Beethoven ...
211	COL   38358   Nebraska   Springsteen ...
256	DG   18807   Symphony No. 9   Beethoven ...
300	MER   75016   Coq d'or Suite   Rimsky ...
353	COL   31809   Symphony No. 9   Dvorak ...
396	DG   139201   Violin Concerto   Beethoven ...
442	FF   245   Good News   Sweet Honey In The ...

*arquivo de índice secundário*

*arquivo de dados*



# Tight Binding

*(fortemente ligado)*

---

- Índices primário e secundário
  - a ligação da chave a um endereço físico no arquivo de dados ocorre à medida que o arquivo de dados é construído (ex.: inserções)





# Tight Binding

## *(fortemente ligado)*

---

- Vantagens

- acesso direto
  - índice primário → *byte offset*
  - índice secundário → *byte offset*
- acesso mais rápido a partir dos índices secundários

- Desvantagens

- alto custo para modificações
  - uma modificação no arquivo de dados afeta tanto o índice primário quanto todos os índices secundários relacionados
- complexidade da codificação e da recuperação de falhas

# Loosely Binding (*fracamente ligado*)

Beethoven	ANG3795
Beethoven	DG139201
Beethoven	DG18807
Beethoven	RCA2626
Corea	WAR23699
Dvorak	COL31809
Prokofiev	LON2312
Rimsky	MER75016
Springsteen	COL38358
Sweet Honey In The	FF245

*arquivo de índice secundário*

ANG3795	167
COL31809	353
COL38358	211
DG139201	396
DG18807	256
FF245	442
LON2312	32
MER75016	300
RCA2626	77
WAR23699	132

*arquivo de índice primário*

32	LON   2312   Rom ...
77	RCA   2626   Quar ...
132	WAR   23699   Tou ...
167	ANG   3795   Sym ...
211	COL   38358   Nebr ...
256	DG   18807   Sym ...
300	MER   75016   Coq ...
353	COL   31809   Sym ...
396	DG   139201   Violin ...
442	FF   245   Good ...

*arquivo de dados*



# Loosely Binding (*fracamente ligado*)

---

- Índice primário
  - a ligação da chave primária a um endereço físico ocorre à medida que o arquivo de dados é construído (ex.: inserções)
- Índice secundário
  - a ligação da chave primária a um endereço físico acontece no momento da pesquisa



# Loosely Binding (*fracamente ligado*)

---

- Vantagens

- diminui custo para as modificações com a abordagem *delete some references*
- uma modificação no arquivo de dados afeta no máximo somente o índice primário
- previne erros de codificação e de falhas do sistema

- Desvantagens

- indireção
  - índice secundário → índice primário → *byte offset*
- queda no desempenho da pesquisa