

# AULA 4- PROCESSAMENTO DE CONSULTAS

PROFA. DRA. LEILA BERGAMASCO

CC6240 – Tópicos Avançados em Banco de Dados

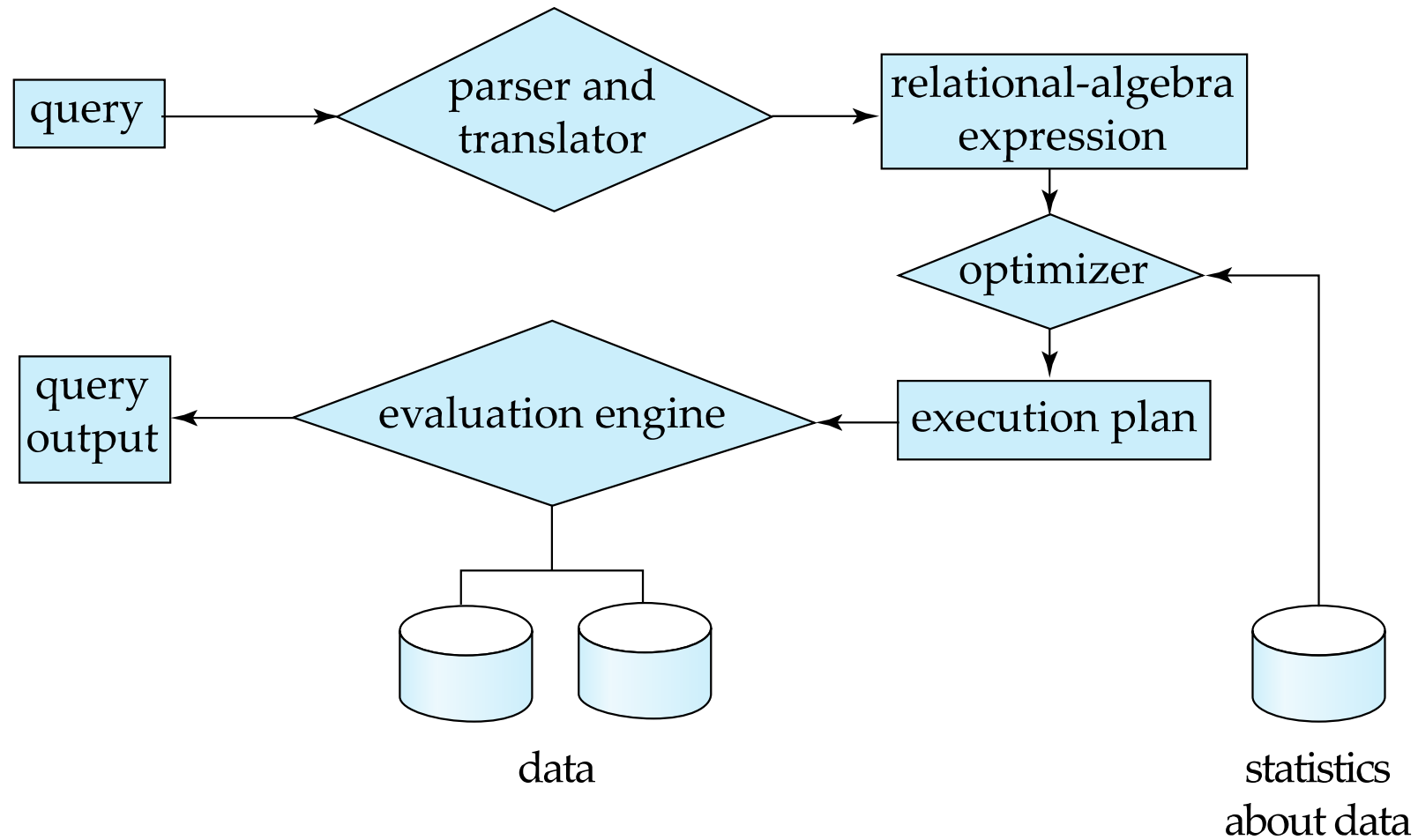
# AGORA VEREMOS

- Como SGBD processa consultas
- Fatores envolvidos no processamento de consultas
- Custos de consulta

# PROCESSAMENTO DE CONSULTAS

- Finalidade
  - Apoiar atividades envolvidas na ação de extrair dados de um BD
- Que atividades são essas?
  - Tradução alto nível → expressões implementadas em nível físico
  - Otimizações
  - Avaliação

# PROCESSAMENTO DE CONSULTAS



# PROCESSAMENTO DE CONSULTAS

## ■ Exemplo de consulta SQL

```
select saldo  
from conta  
where saldo < 2500
```

## ■ Como fica em Álgebra Relacional?

$$\Pi_{saldo}(\sigma_{saldo < 2500}(Conta))$$
$$\sigma_{saldo < 2500}(\Pi_{saldo}(Conta))$$

Mais de uma forma de representação algébrica.

Para cada representação vários algoritmos

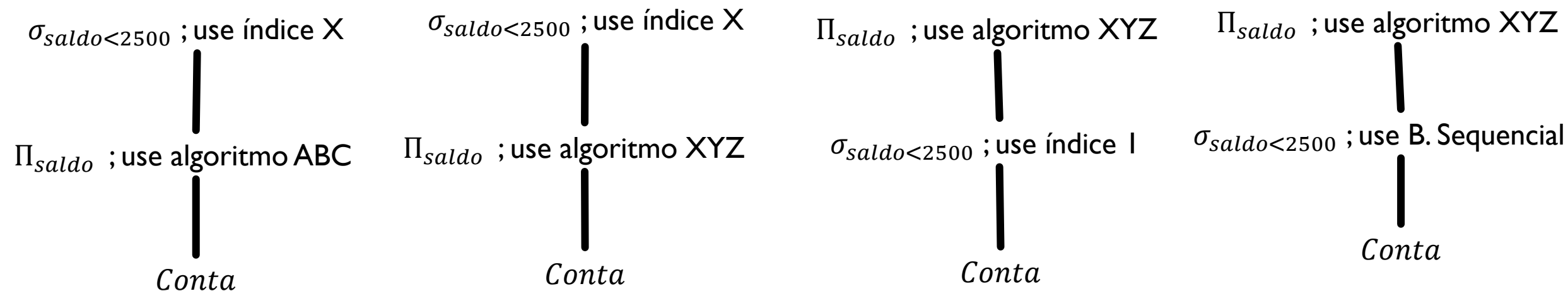
Usando o que aprendemos com índices, como  
pode ser feita a seleção?

# PROCESSAMENTO DE CONSULTAS

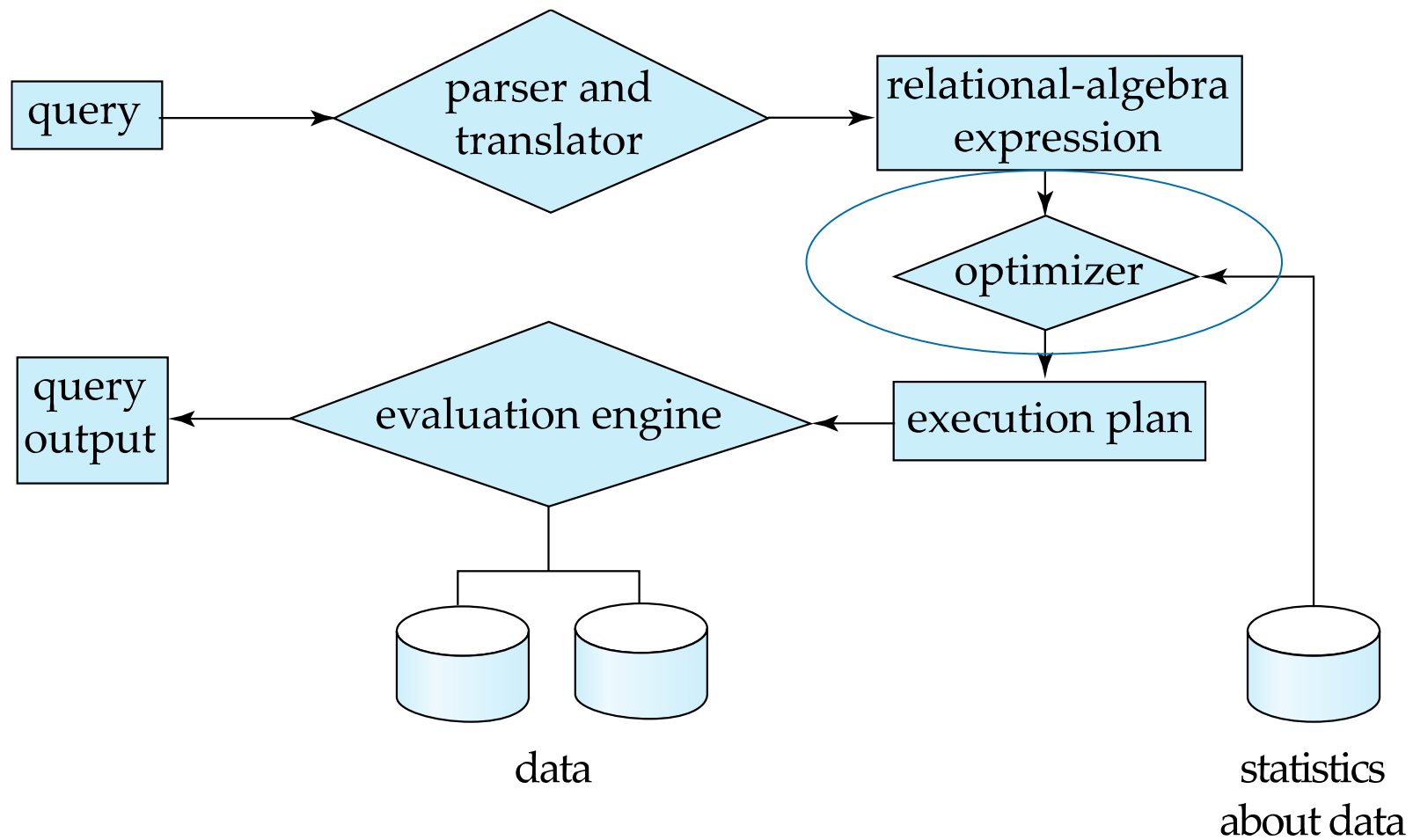
- SGBDs trabalham para minimizar custos
- E o que é considerado? Qual medida de desempenho mais crítica?  
NÚMERO DE ACESSO A DISCOS

■ Para avaliar uma consulta, não basta expressão algébrica relacional. É necessário:

- Anotar para cada operação como iremos avaliá-la (instruções)
  - Utilizar índices? Algoritmo de busca sequencial?
- Expressão algébrica relacional + instruções sobre como ser avaliada = avaliação primitiva (AP)
- Várias APs podem ser agrupadas em pipeline, permitindo execuções em paralelo
- Sequência de APs é chamada de plano de execução de consulta (PEC).



# PROCESSAMENTO DE CONSULTAS

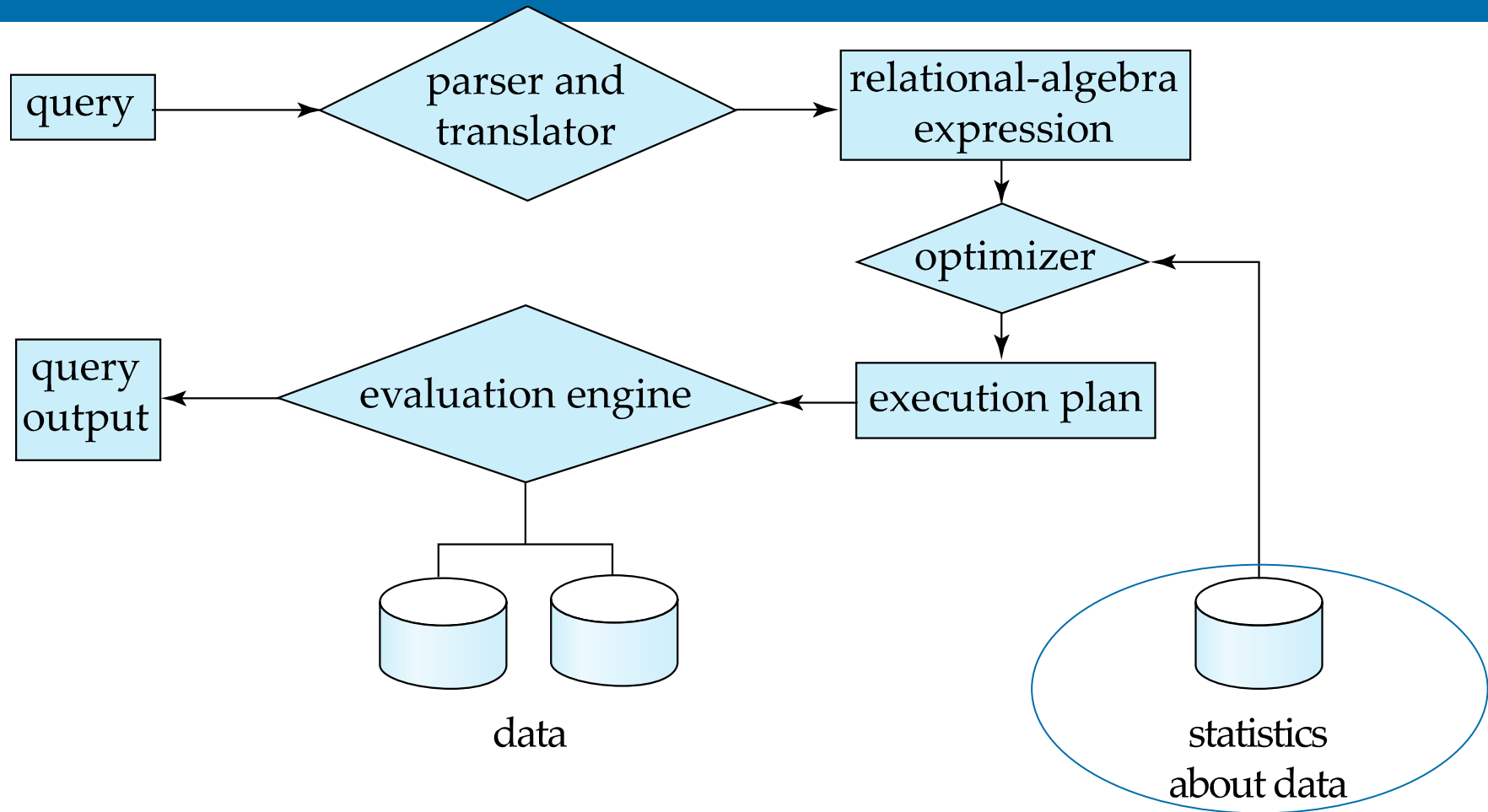


# O QUE É UMA OTIMIZAÇÃO DE CONSULTAS

- Então, o que é a otimização de consultas?
  - processo de selecionar o plano de avaliação de consulta mais eficiente para uma consulta.
- Possibilidades:
  - achar expressão equivalente em álgebra relacional que seja mais eficiente (veremos mais adiante).
  - estratégia adequada para processar consulta (algoritmos, índices).
- Como escolher melhor plano de avaliação de consulta?
  - não é possível calcular custo exato sem executar plano
    - Você pode então estimar com base em algumas estatísticas



# PROCESSAMENTO DE CONSULTAS



## COMO ESTIMAR O CUSTO DE UMA PEC?

- Consultando as estatísticas:
  - $n_r$ - número de tuplas na relação  $r$
  - $b_r$  – número de blocos que contêm tuplas da relação  $r$
  - $s_r$ – tamanho em bytes de uma tupla da relação  $r$
  - $f_r$ – fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco
  - $V(A,r)$  – número de valores distintos para o atributo  $A$  na relação  $r$
  - $SC(A,r)$  – número médio de registros que satisfazem uma condição de igualdade no atributo  $A$
- $f_i$  – fanout médio dos nós internos do índice  $i$  (número de ponteiros em um nó)
- $HT_i$ – número de níveis no índice  $i$  (altura da árvore). Se árvore balanceada,  $HT_i = \lceil \log_{f_i} (V(A,r)) \rceil$
- $LB_i$  – número de blocos de índice de nível mais baixo no índice  $i$

# COMO ESTIMAR O CUSTO DE UMA PEC?

- Problema das estatísticas

- Manutenção

- overhead devido à atualização dos dados estatísticos toda vez que houver alteração na relação ou no índice
    - solução: atualização periódica

# COMO ESTIMAR O CUSTO DE UMA PEC?

- Além das estatísticas precisamos saber os possíveis algoritmos utilizado para cada operação utilizada.
- Chamamos de  $E_{A_i}$  o custo do algoritmo  $A_i$

Operação	Algoritmo
Seleção	A1 (busca linear)
	A2 (busca binária)
	A3 (índice primário, igualdade de chave)
	A4 (índice primário, igualdade em atributo não chave)
	A5 (índice secundário, igualdade)
	A6 (índice primário, comparação)
	A7 (índice secundário, comparação)
	A8 (seleção de conjunção usando um índice)
	A9 (seleção de conjunção usando índice composto)
	A10 (seleção de conjunção usando intersecção de identificadores)
	A11 (seleção de disjunção usando união de identificadores)
Classificação	sort-merge externo
Junção	Junção de laço aninhado
	Junção de laço aninhado de blocos
	Junção de laço aninhado indexada
	Merge-junção
	Hash-Junção

# OPERAÇÃO DE SELEÇÃO

num_agencia	nome_agencia
123	Campinas
456	São Bernardo do Campo
555	São Paulo
216	Santos

## ■ AI (busca linear)

- varre cada bloco do arquivo e todos registros são testados para verificar a condição de seleção
- todos blocos precisam ser lidos.

- Custo:  $E_{A_1} = b_r$

- se seleção for em atributo chave\*, estima-se que metade dos blocos é varrida

- Custo:  $E_{A_1} = \frac{b_r}{2}$

```
SELECT * FROM BANCO  
WHERE num_agencia = 123
```

- Em geral é o mais lento, porém pode ser aplicado em qualquer tipo de arquivo, com índice ou não.

\* não há chance de encontrarmos mais de um registro em outros blocos

$$SC(A, r) = \frac{n_r}{V(A, r)}$$

num_agencia	nome_agencia
123	Campinas
456	São Bernardo do Campo
555	São Paulo
216	Santos

## OPERAÇÃO DE SELEÇÃO

Custo para achar 1ª  
tupla

$$E_{A_2} = \lceil \log_2(b_r) \rceil + \left\lceil \frac{SC(A, r)}{f_r} \right\rceil - 1$$

Quantidade total de registros que satisfazem a condição dividido pela quantidade total de tuplas que cabem no bloco, dos quais um já havia sido recuperado (por isso o -1).

### ■ A2 (busca binária)

- Arquivo ordenado fisicamente pelo atributo usado na seleção (igualdade):

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - número médio de registros que satisfazem uma condição de igualdade no atributo  $A$

```
SELECT * FROM BANCO
WHERE nome_agencia = Santos
```

# OPERAÇÃO DE SELEÇÃO

num_agencia	nome_agencia
123	Campinas
216	Santos
456	São Bernardo do Campo
555	São Paulo

Custo para achar 1ª  
tupla

$$E_{A_2} = \lceil \log_2(b_r) \rceil$$

## ■ A2 (busca binária)

- Arquivo ordenado fisicamente pelo atributo usado na seleção (igualdade) e é chave primária

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A,r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A,r)$  - número médio de registros que satisfazem uma condição de igualdade no atributo  $A$

```
SELECT * FROM BANCO
WHERE num_agencia = 123
```

Conseguem  
perceber a diferença  
do A1?

# OPERAÇÃO DE SELEÇÃO

$\sigma_{nome-agencia="Campinas"}(Conta)$

$f_{conta}=20$

$V(nome\_agencia, conta) = 50$

$n_{conta}=10000$

Usando A1: busca sequencial

$$b_r = \frac{10000}{20} = 500$$

$$E_{A_1} = b_r$$

Usando A2: busca binária

$$SC = \frac{10000}{50} = 200$$

$$\frac{200}{20} = 10$$

$$\log_2 500 = 9$$

$$E_{A_2} = \lceil \log_2(b_r) \rceil + \left\lceil \frac{SC(A, r)}{f_r} \right\rceil - 1$$

$$9 + 10 - 1 = 18 \text{ acessos a blocos}$$

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$

(número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

<u>num_agencia</u>	<u>nome_agencia</u>
123	Campinas
456	São Bernardo do Campo
555	São Paulo
216	Santos



# OPERAÇÃO DE SELEÇÃO

num_agencia	nome_agencia
123	Campinas
456	São Bernardo do Campo
555	São Paulo
216	Santos

## ■ A3 (índice primário, igualdade de chave única)

$$\sigma_{num-agencia=123}(Agencia) \xrightarrow{E_{A_3} = HT_i + 1}$$

Índice criado em chave primária e igualdade é nesse atributo

$$f_{conta}=20$$

$$V(num\_agencia, conta) = 10000$$

$$n_{agencia}=10000$$

$$f_i = 40 \text{ (arvore B+)}$$

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r))$$

$$HT_i = \log_{\frac{40}{2}}(10000) = 4$$

$$E_{a_3} = 4 + 1 = 5$$

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$  (número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  - *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  - número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

$LB_i$  - número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

num_agencia	nome_agencia
123	Campinas
217	Santos
216	Santos

## ■ A4 (índice primário, igualdade em atributo não chave única)

$$E_{A_4} = HT_i + \left\lceil \frac{SC(A,r)}{f_r} \right\rceil$$

$\sigma_{nome-agencia=Santos}(Agencia)$

$f_{conta}=20$

$V(cidade-agencia, conta) = 50$

$n_{agencia}=10000$

$f_i = 40$  (arvore B+)

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r)) \quad E_{A_4} = 2 + \left\lceil \frac{200}{20} \right\rceil = 2 + 10 = 12$$

$$HT_i = \log_{\frac{40}{2}}(50) = 2$$

$$SC(A, r) = \frac{10000}{50} = 200$$

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$  (número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  - *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  - número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

$LB_i$  - número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

<u>num_agencia</u>	<u>nome_agencia</u>
123	Campinas
217	Santos
216	Santos

## ■ A5 (índice secundário, igualdade)

- Recupera 1 registro se campo indexação for uma chave única

$\sigma_{num-agencia=123}(Agencia)$

$f_{conta}=20$

$V(cidade-agencia, conta) = 50$

$n_{agencia}=10000$

$f_i = 40$  (arvore B+)

$$E_{A_3} = HT_i + 1$$

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r))$$

$$HT_i = \log_{\frac{40}{2}}(50) = 2$$

$$E_{A_5} = 2 + 1 = 3$$

Arquivo **não** está organizado por num-agencia, porém é atributo chave e segue as mesmas regras que A3

$n_r$ - número de tuplas na relação  $r$

$b_r$  – número de blocos que contêm tuplas da relação  $r$

$s_r$  – tamanho em bytes de uma tupla da relação  $r$

$f_r$  – fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  – número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  – cardinalidade da seleção do atributo  $A$  da relação  $r$  (número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  – *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  – número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

$LB_i$  – número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

<u>num_agencia</u>	<u>nome_agencia</u>
789	Xangai
217	Santos
123	Campinas
216	Santos

## ■ A5 (índice secundário, igualdade)

- Recupera vários registros se campo indexação não for chave única → Pior caso: cada registro que satisfaz condição em 1 bloco diferente

$\sigma_{nome-agencia=Santos}(Agencia)$

$f_{conta}=20$

$V(cidade-agencia, conta) = 50$

$n_{agencia}=10000$

$f_i = 40$  (arvore B+)

$$E_{A_5} = HT_i + SC(A, r)$$

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r))$$

$$HT_i = \log_{\frac{40}{2}}(50) = 2$$

$$SC(A, r) = \frac{10000}{50} = 200$$

$$E_{A_5} = 2 + 200 = 202$$

Qual a diferença  
para o A4?

Arquivo **não** está  
organizado por  
nome\_agencia e  
ela não é unica

$n_r$ - número de tuplas na relação  $r$

$b_r$  – número de blocos que contêm tuplas da relação  $r$

$s_r$  – tamanho em bytes de uma tupla da relação  $r$

$f_r$  – fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  – número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  – cardinalidade da seleção do atributo  $A$  da relação  $r$   
(número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  – *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  – número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

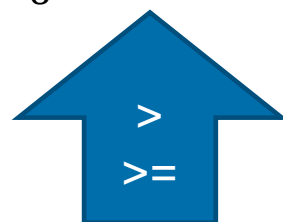
$LB_i$  – número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

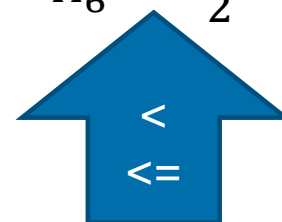
$$\sigma_{A \leq v}(r)$$

- A6 (índice primário, comparação)
  - suposição: metade dos registros satisfaz a condição
    - resultado tem o número de tuplas  $= \frac{b_r}{2}$
- Performance diferentes com operadores diferentes ( $\leq$  ou  $\geq$ )
  - condição do tipo  $A \geq v$ : encontra primeiro registro que atende a condição e varre arquivo até final
  - condição do tipo  $A \leq v$ : varre do início do arquivo até encontrar tupla com valor  $\geq v$ .

$$E_{A_6} = HT_i + \frac{b_r}{2}$$



$$E_{A_6} = \frac{b_r}{2}$$



## OPERAÇÃO DE SELEÇÃO

<u>num_agenci</u> <u>a</u>	nome_agencia
123	Campinas
217	Santos
216	Santos

### ■ A6 (índice primário, comparação)

$\sigma_{nome-agencia \leq Santos}(Agencia)$

$f_{conta} = 20$

$V(nome-agencia, conta) = 50$

$n_{agencia} = 10000$

índice armazena 20 ponteiros por nós

índice árvore-B primário para atributo nome\_agencia

$$E_{A_6} = \frac{b_r}{2}$$

$$b_r = \frac{10000}{20} = 500$$

$$E_{A_6} = \frac{500}{2} = 250$$

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$

(número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  - *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  - número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i} (V(A, r)) \rceil$

$LB_i$  - número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

<u>num_agencia</u>	<u>nome_agencia</u>
123	Campinas
217	Santos
216	Santos

## ■ A6 (índice primário, comparação)

$\sigma_{nome-agencia \geq Campinas}(Agencia)$

$f_{conta} = 20$

$V(nome-agencia, conta) = 50$

$n_{agencia} = 10000$

índice armazena 20 ponteiros por nós

índice árvore-B primário para atributo nome\_agencia

$$E_{A_6} = HT_i + \frac{b_r}{2}$$

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r))$$

$$HT_i = \log_{\frac{20}{2}}(50) = 2$$

$$b_r = \frac{10000}{20} = 500$$

$$E_{A_6} = 2 + \frac{500}{2} = 252$$

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$

(número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$f_i$  - *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  - número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

$LB_i$  - número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

## OPERAÇÃO DE SELEÇÃO

<u>num_agencia</u>	<u>nome_agencia</u>
123	Campinas
217	Santos
216	Centro

- A7 (índice secundário, comparação)
  - blocos de índice de nível mais baixo são varridos a partir do maior ( $\geq$ ) ou menor valor ( $\leq$ ) até v.
  - suposição: metade dos registros satisfaz a condição
 
$$E_{A_7} = HT_i + \frac{LB_i}{2} + \frac{n_r}{2}$$
    - Quantos blocos de índice de nível mais baixo serão acessados? Metade!
    - E ainda: caminho percorrido no índice a partir da raiz do bloco até primeiro bloco folha ser usado.

$f_i$  – fanout médio dos nós internos do índice  $i$  para índices estruturados em árvore  
 $HT_i$  – número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i} (V(A,r)) \rceil$   
 $LB_i$  – número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)



# OPERAÇÃO DE SELEÇÃO

## ■ A7 (índice secundário, comparação)

$\sigma_{nome\_agencia < Santos}(Conta)$

$f_{conta} = 20$

$V(saldo, conta) = 500$

$n_{agencia} = 10000$

índice armazena 20 ponteiros por nós

índice árvore-B **secundário** para atributo saldo

$n_r$  - número de tuplas na relação  $r$

$b_r$  - número de blocos que contêm tuplas da relação  $r$

$s_r$  - tamanho em bytes de uma tupla da relação  $r$

$f_r$  - fator de bloco da relação  $r$ : número de tuplas da relação  $r$  que cabe em um bloco

$V(A, r)$  - número de valores distintos para o atributo  $A$  na relação  $r$

$SC(A, r)$  - cardinalidade da seleção do atributo  $A$  da relação  $r$

(número médio de registros que satisfazem uma condição de igualdade no atributo  $A$ )

$$E_{A_7} = HT_i + \frac{LB_i}{2} + \frac{n_r}{2}$$

$$HT_i = \log_{\frac{f_i}{2}}(V(A, r))$$

$$HT_i = \log_{\frac{20}{2}}(500) = 3$$

$$LB_i = \frac{V(A, r)}{f_i} = \frac{500}{20} = 25$$

$$E_{A_7} = 3 + 25 + \frac{10000}{2} = 5028 \text{ acessos!}$$

E se usássemos busca  
sequencial?  
 $10000/20 = 500$  acessos!

$f_i$  - *fanout* médio dos nós internos do índice  $i$  para índices estruturados em árvore

$HT_i$  - número de níveis no índice  $i$  (altura do índice). Se árvore balanceada,  $HT_i = \lceil \log_{f_i}(V(A, r)) \rceil$

$LB_i$  - número de blocos de índice de nível mais baixo no índice  $i$  (nível de folha)

# OPERAÇÃO DE SELEÇÃO

- Seleções Complexas (Conjunção, Disjunção, Negação)
  - Deve-se calcular o custo de cada seleção
  - Adotar a conduta para cada tipo de seleção
  - Algoritmos específicos para cada tipo de seleção

$$\sigma_{nome-agencia=Centro \wedge saldo < 1500}(\text{Conta})$$

Conjunção - AND

$$\sigma_{\theta_1 \wedge \theta_2 \wedge \theta_3}$$

Disjunção - OR

$$\sigma_{\theta_1 \vee \theta_2 \vee \theta_3}$$

$$\sigma_{A=v \wedge A < x}(r)$$

## OPERAÇÃO DE SELEÇÃO

- A8 (seleção de conjunção usando um índice)
  - determinar se há um caminho de acesso para um dos atributos em uma das condições simples
    - escolher um dos algoritmos (A1 a A7) para recuperar registros daquela condição
    - completar operação testando se cada registro no buffer de memória satisfaz as demais condições simples
  - Custo: Como leva para o buffer de memória utilizando índice, o custo é o próprio algoritmo. Após isso, é comparado com os demais atributos da condição.

## OPERAÇÃO DE SELEÇÃO

$$\sigma_{A=v \wedge A=x}(r)$$

- A9 (seleção de conjunção usando índice composto)
  - Condição de igualdade
  - Pode-se procurar diretamente no índice se a seleção atribui uma condição de igualdade em dois ou mais atributos e se ainda existe um índice composto nesses campos de atributos combinados.
  - O tipo de índice determina qual algoritmo, A3, A4 ou A5 vai ser utilizado.
  - Custo = melhor caso

## OPERAÇÃO DE SELEÇÃO

- A10 (seleção de conjunção usando intersecção de identificadores)
  - Não recupera o registro e sim o ponteiro para o registro.
  - Intersecção entre ponteiros recuperados fornece resultado.
  - Custo adicional de leitura e intersecção

# OPERAÇÃO DE SELEÇÃO

- AII (seleção de disjunção usando união de identificadores)
  - cada índice é varrido buscando-se os ponteiros que satisfazem a condição
  - união entre ponteiros recuperados fornece resultado.
  - Custo = pior caso.

## BIBLIOGRAFIA

- ABRAHAM SILBERSCHATZ, HENRY F. KORTH, S. SUDARSHAN. Sistema de Banco de Dados. 6. Campus. 0. ISBN 9788535245356.
- ELMASRI, RAMEZ, SHAMKANT B. NAVATHE. Sistemas de banco de dados. Vol. 6. São Paulo: Pearson Addison Wesley, 2011.
- DATE, CHRISTHOPER J. Introdução a Sistemas de Bancos de Dados, 5ª. Edição. Campus, Rio de Janeiro (2004).

OBRIGADO E ATÉ A PRÓXIMA AULA!

