

Arquitetura de SGBD Relacionais — Previsão de Seletividade —

Caetano Traina Jr.

Grupo de Bases de Dados e Imagens
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo - São Carlos
caetano@icmc.usp.br

20 de abril de 2013
São Carlos, SP - Brasil

Apresentam-se os conceitos de seletividade de predicados e as propriedades dos atributos e dos tipos de predicados que habilitam a estimativa de custo de operadores de acesso físico.

Outline

- 1 Conceitos básicos
- 2 Métricas de Previsão de Seletividade
- 3 Distribuição de Valores de Atributos
- 4 Predicados indexáveis
- 5 Uso de índices para responder a consultas
- 6 Coleta de estatísticas em SQL

Previsão de Seletividade

Introdução

Considere-se a operação de seleção:

$$\text{Resultado} = \sigma_{\langle \text{predicado} \rangle} \text{Relação}$$

- Operadores de seleção e de junção usam o predicado associado como um **filtro** sobre o conjunto de tuplas que são submetidas ao operador, para obter o subconjunto resultado que atende ao predicado.
- A proporção de tuplas que fazem parte do resultado é chamado de **Fator de Seletividade do predicado**,
- e é definido como

$$Sel(\text{Predicado}) = 1 - Sai = \frac{\text{Número de tuplas no resultado}}{\text{Número de tuplas na entrada}}$$
- Assim, valores de seletividade próximos a um indicam que o predicado é bastante seletivo, pois poucas tuplas conseguem passar pelo “filtro”;
- valores próximos a zero indicam que poucas tuplas foram filtradas.

Previsão de Seletividade

Introdução

- A previsão de seletividade depende de métricas mantidas pelo Gerenciador, e inclui informação sobre:
 - Medidas sobre a Base de dados,
 - Medidas sobre as Tabelas,
 - Medidas sobre cada Atributo em cada tabela,
 - Medidas sobre os índices.
- A maioria das métricas são coletadas sobre os dados armazenados, sendo então chamadas **Estatísticas**.
- As estatísticas são obtidas:
 - durante cada operação de consulta,
 - e pela execução de comandos específicos para coletá-las
 - automaticamente,
 - ou por solicitação do DBA.

Previsão de Seletividade

Métricas

A métrica fundamental vem da:

- A base de dados:
 - O Tamanho da página da base (*PAGESIZE*)

O tamanho da página tem que ser constante entre todos os segmentos, extents, páginas, esquemas da base, etc., porque todas compartilham a mesma estrutura de *cache* do gerenciador.

Previsão de Seletividade

Estatísticas

Além dela, existem estatísticas sobre:

- Cada tabela:
 - Tamanho médio de tupla em disco ($ROWSIZE = ?$),
 - Número de tuplas gravadas ($CARD = 10.000$),
 - Número máximo de tuplas já existentes ($MAXCARD = 10.000$),
 - Número de páginas gravadas ($NPAGS = \lceil 1 + CARD/20 \rceil$),
- Cada atributo em cada tabela:
 - Tamanho médio do atributo em disco ($COLSIZE = ?$),
 - Tamanho do domínio ativo ($COLCARD = 25$),
 - Distribuição de valores ($DISTR = 1$),
 - Segundo menor valor nesse atributo ($LOW2KEY = ?$),
 - Segundo maior valor nesse atributo ($HIGH2KEY = ?$)

Previsão de Seletividade

Estatísticas

- Cada índice criado:
 - Número de níveis numa B-tree ($H = 0$),
 - Número de nós-folha no índice ($NLEAFS = CARD/300$),
 - Tamanho do domínio ativo do primeiro atributo indexado ($KEY1CARD = 25$),
 - Tamanho do domínio ativo da chave inteira ($KEYFULLCARD = 25$),
 - Porcentagem de tuplas clusterizadas ($CLUSTERRATIO = 0$ if not clustered, = 95 otherwise)
- ...

Tipos de acesso aos objetos

Exemplo

- Por exemplo, seja a relação de alunos:

```
CREATE TABLE Alunos (  
    NUSP CHAR(10) PRIMARY KEY,  
    Nome VARCHAR(50),  
    Idade DECIMAL(3),  
    Cidade VARCHAR(40) );
```

- Veja que o tamanho máximo da tupla seria $10 + 50 + \frac{3}{2} + 40 = 102$ bytes.
- Como nem todos os alunos têm 50 letras no nome, nem as cidade têm 40 letras, a média deve ser menor que isso. Vamos assumir que a média seja **74** bytes.

Previsão de Seletividade

Estatísticas – Exemplo

- Vamos considerar que existam **80.000 alunos** mas já chegou a um máximo de **83.000**, oriundos de **700** cidades distintas, variando de **15 a 85 anos em 50** idades distintas e que **1.000** deles tenham nomes repetidos.
- Inicialmente vamos assumir que o SGBD está operando nessa tabela com todos os atributos em distribuição Constante `DISTR=1`. Então as estatísticas serão algo assim:

Métrica da base de Dados: `PAGESIZE=2.048`

Métricas para Tabela:

Nome	RowSIZE	CARD	MaxCARD	NPags
Alunos	?	10.000	10.000	500
...		...		

Métricas para Atributos:

Tabela	Atrib	ColSIZE	ColCARD	Distr	Low2Key	high2Key
Alunos	NUSP	10	25	1	?	?
Alunos	Nome	?	25	1	?	?
Alunos	Idade	2	25	1	?	?
Alunos	Cidade	?	25	1	?	?

Previsão de Seletividade

Estatísticas – Exemplo

- Vamos considerar que existam **80.000 alunos** mas já chegou a um máximo de **83.000**, oriundos de **700** cidades distintas, variando de **15 a 85 anos em 50** idades distintas e que **1.000** deles tenham nomes repetidos.
- A seguir, vamos assumir que o SGBD está operando com essa tabela com todos os atributos em distribuição Univorme `DISTR=2`. Então as estatísticas serão algo assim:

Métrica da base de Dados: `PAGESIZE=2.048`

Métricas para Tabela:

Nome	RowSIZE	CARD	MaxCARD	NPags
Alunos	74	80.000	83.000	3.810
...		...		

Métricas para Atributos:

Tabela	Atrib	ColSIZE	ColCARD	Distr	Low2Key	high2Key
Alunos	NUSP	10	80.000	2	'0011223344'	'9988776655'
Alunos	Nome	25	79.000	2	'Abraão Abraão'	'Zuleica Penultima'
Alunos	Idade	2	50	2	17	80
Alunos	Cidade	18	700	2	'Adamantina'	'Zuliana'

Distribuição de Valores de Atributos

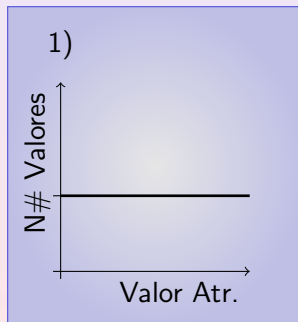
Métricas

Métricas sobre a distribuição de valores de cada atributo em cada tabela:

1 – Constante:

Assume um valor constante para a seletividade.

Independente de quantos valores o atributo tenha, considera-se que cada valor de atributo, ou cada faixa de valores de atributos ocorra $1 - \frac{1}{\text{COLCARD}}$ vezes.



Distribuição de Valores de Atributos

Métricas

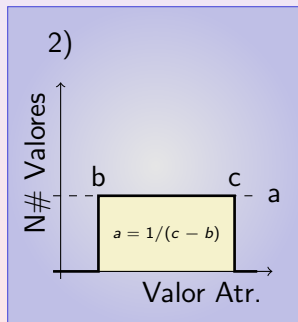
Métricas sobre a distribuição de valores de cada atributo em cada tabela:

2 – Distribuição Uniforme:

Cada valor contribui com a saída proporcional de $\frac{1}{|\text{Domínio ativo}|}$.

Aqui define-se a faixa de valores mínima e máxima que cada atributo assume.

Assim, cada valor de atributo ocorre proporcional a $\frac{1}{\text{COLCARD}}$ vezes em relação à faixa inteira de valores de atributos, levando à seletividade de $1 - \frac{1}{\text{COLCARD}}$.



Distribuição de Valores de Atributos

Métricas

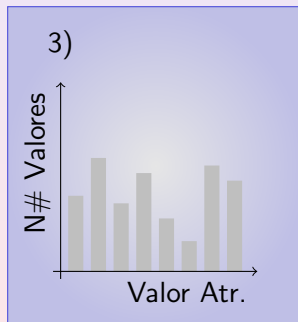
Métricas sobre a distribuição de valores de cada atributo em cada tabela:

3 – Histograma:

Deve ser mantido um contador para cada valor do domínio ativo (Arquivo Invertido).

Esta distribuição é feita para atributos discretos com domínios de pequena cardinalidade (< 1.000).

Os atributos em geral são indexados como arquivos invertidos ou Bitmap.



Distribuição de Valores de Atributos

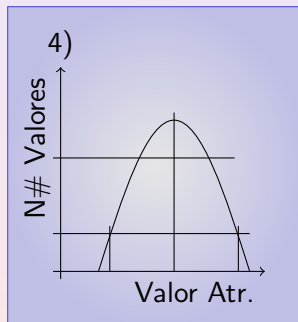
Métricas

Métricas sobre a distribuição de valores de cada atributo em cada tabela:

4 – Distribuição Estatística:

Armazena menor e maior valor do domínio ativo, mais dados sobre a distribuição estatística assumida (Gaussiana, Exponencial, etc.).

Esta distribuição é feita para atributos numéricos contínuos em domínios de cardinalidade grande.



Distribuição de Valores de Atributos

Seletividade de predicados

- Note-se que a seletividade é calculada sempre sobre um **predicado** $pred$.
- Todo predicado é da forma $pred = atr_1 \theta atr_2$ ou $pred = atr_1 \theta cte$.
- Assim, o cálculo da seletividade pode usar as métricas do atributo atr_1 , e pode usar a distribuição de valores prevista para esse atributo.
- Considerando que uma distribuição está definida para um determinado índice, atributo ou relação, a seletividade de um predicado, indicada como $SEL(< pred >)$ pode ser calculada pelo tipo do predicado envolvendo o atributo.

Distribuição de Valores de Atributos

Seletividade de predicados

Para distribuição uniforme, a seletividade é calculada usando a proporção de saída de atributos $Sai(pred)$:

Tipo de predicado	$Sai(< pred >)$
$Atr = cte$	$1 / COLCARD$
$Atr \theta cte$	Combinação, por exemplo:
$Atr < cte$	$\frac{\max(cte - LOW2KEY, 0)}{HIGH2KEY - LOW2KEY}$
$Atr BETWEEN cte_1 AND cte_2$	$\frac{ cte_2 - cte_1 }{HIGH2KEY - LOW2KEY}$
$Atr IN list$	$\frac{ list }{COLCARD}$
$Atr IS NULL$	$1 / COLCARD$
$Atr LIKE ('pattern\%')$	Interpolação, depende do alfabeto
$Atr LIKE ('\%pattern')$	$1 / COLCARD$
$pred_1 AND pred_2$	$Sai(pred_1) * Sai(pred_2)$
$pred_1 OR pred_2$	$Sai(pred_1) + Sai(pred_2) - Sai(pred_1) * Sai(pred_2)$
$NOT pred$	$1 - Sai(pred_1)$

Distribuição de Valores de Atributos

Seletividade de predicados – Exemplo

- Por exemplo, seja a relação de alunos do exemplo anterior, com as respectivas métricas.
- Inicialmente, vamos assumir que o SGBD está operando nessa relação com todos os atributos em distribuição constante $\boxed{\text{DISTR}=1}$.

Seja a consulta:

```
SELECT *
  FROM Alunos
 WHERE Nome='José da Silva';
```




A Seletividade será


$$\text{Sel}(\text{Nome}='José da Silva') = 1 - 1/\text{COLCARD} = 1/25 = 0,96.$$


Distribuição de Valores de Atributos


Seletividade de predicados – Exemplo


- Variando o predicado da cláusula **WHERE** teríamos por exemplo:

 $Sel(\text{Nome}='José da Silva') = 1 - \frac{1}{COLCARD} = 1/25 = 0,96.$

 $Sel(\text{Idade}=20) = 1 - \frac{1}{COLCARD} = 1/25 = 0,96.$

 $Sel(\text{Idade}<20) = 1 - \frac{1}{COLCARD} = 1/25 = 0,96.$

 $Sel(\text{Idade}<10) = 1 - \frac{1}{COLCARD} = 1/25 = 0,96.$

 $Sel(\text{Idade BETWEEN 20 AND 30}) = 1 - \frac{1}{COLCARD} = 1/25 = 0,96.$

Distribuição de Valores de Atributos

Seletividade de predicados – Exemplo

- Com o SGBD operando com todos os atributos dessa relação em distribuição univorme $\boxed{\text{DISTR}=2}$ teríamos por exemplo:

$$\text{Sel}(\text{Nome}='José da Silva') = 1 - \frac{1}{\text{COLCARD}} = 1 - \frac{1}{79.000} \approx 1,0.$$

$$\text{Sel}(\text{Idade}=20) = 1 - \frac{1}{\text{COLCARD}} = 1 - \frac{1}{50} = 0,98.$$

$$\text{Sel}(\text{Idade}<20) = \frac{\max(\text{cte}-\text{LOW2KEY}, 0)}{\text{HIGH2KEY}-\text{LOW2KEY}} = 1 - \frac{\max(20-17, 0)}{80-17} = 1 - \frac{3}{63} = 0,9524.$$

$$\text{Sel}(\text{Idade}<10) = \frac{\max(\text{cte}-\text{LOW2KEY}, 0)}{\text{HIGH2KEY}-\text{LOW2KEY}} = 1 - \frac{\max(10-17, 0)}{80-17} = 0.$$

$$\text{Sel}(\text{Idade BETWEEN 20 AND 30}) = \frac{|\text{cte}_2 - \text{cte}_1|}{\text{HIGH2KEY}-\text{LOW2KEY}} = 1 - \frac{|30-20|}{80-17} = 0,8413.$$

Predicados Indexáveis

- Índices podem ser usados para ajudar a filtrar as tuplas que fazem parte do resultado.
- Índices tendem a ser úteis para acessar os dados quando sua seletividade é grande, tipicamente maior que 90%.
- Portanto, um índice torna-se útil quando for usado para filtrar um predicado com seletividade elevada. A decisão pelo uso de um índice é baseada no cálculo do predicado que pode ser feito pelo uso do índice.
- Um índice pode ser usado por dois motivos:
 - Para localizar as tuplas (*matching*);
 - Para restringir as tuplas (*screening*);
- Um índice poderá ser usado para agilizar uma busca se houver na consulta um **predicado indexável** por aquele índice.

Predicados Indexáveis

- Suponha que uma consulta Q envolve os predicados $p_1 \wedge p_2 \wedge \dots$ respectivamente sobre os atributos $atr_1, atr_2 \dots$

- Dizemos que

“o predicado p_i é indexável pelo índice idx para a consulta Q ”

quando:

- 1 o índice idx pode ser pesquisado para localizar as tuplas identificadas por esse predicado;
- 2 esse predicado emprega:
 - o primeiro atributo do índice,
 - ou todos os atributos anteriores a a_i ; nesse índice também ocorrem na consulta Q ;
- 3 e o predicado é de tipo indexável.

Predicados Indexáveis

Tipos de Predicados Indexáveis

Quando o predicado é de tipo indexável:

Tipo de predicado	Indexável	Nota
$Atr\theta cte$	Sim	$\theta \in \{<, \leq, =, \geq, >\}$ mas não \neq
Atr BETWEEN cte_1 AND cte_2	Sim	termina uma lista de atributos
Atr IN $list$	Sim	Só pode ser usado uma vez por índice
Atr IS NULL	Sim	
Atr LIKE (' <i>pattern</i> %')	Sim	O % tem que ser o último caracter
Atr LIKE ('% <i>pattern</i> ')	Não	
$Atr_1\theta Atr_2$	Não	atr_1 e atr_2 são da mesma relação
$Atr\theta$ Expressão	Não	
$pred_1$ AND $pred_2$	Sim	Desde que $pred_1$ e $pred_2$ sejam indexáveis
$pred_1$ OR $pred_2$	Não	Exceto ($a_1 = c_1$ OR $a_2 = c_2$)
NOT $pred$	Não	

Predicados de comparação $\theta \in \{<, \leq, \geq, >\}$, LIKE e BETWEEN são chamados de predicados por faixa (*range predicates*).

Predicados de comparação IN *list* ou ($a_1 = c_1$ OR $a_2 = c_2$) são chamados predicados por lista (*IN-list*).

Predicados Indexáveis

Predicados Indexáveis – Exemplo

Seja a relação $\text{Alunos} = \{\text{NUSP}, \text{Nome}, \text{Idade}, \text{Cidade}\}$ para o qual foram criados os índices:

Index PK: {NUSP}

Index NomeNUSP: {Nome, NUSP}

Index CidadeIdade: {Cidade, Idade}

Index IdadeCidadeNome: {Idade, Cidade, Nome}

- Dada a consulta:

```
SELECT * FROM Alunos
      WHERE Nome='José da Silva';
```

- ☞ O predicado $\text{Nome} = \text{'José da Silva'}$ é de tipo indexável, e é indexável por NomeNUSP porque Nome é o primeiro atributo do índice;
- ☞ não é indexável por IdadeCidadeNome porque Nome segue a um atributo não indexável pelo índice;
- ☞ e não é indexável por PK nem por CidadeIdade porque Nome não é indexado nesses índices.

Predicados Indexáveis

Predicados Indexáveis – Exemplo

Seja a relação $\text{Alunos} = \{\text{NUSP}, \text{Nome}, \text{Idade}, \text{Cidade}\}$ para o qual foram criados os índices:

Index PK: {NUSP}




Index NomeNUSP: {Nome, NUSP}

Index CidadeIdade: {Cidade, Idade}

Index IdadeCidadeNome: {Idade, Cidade, Nome}

- Dada a consulta:

```
SELECT * FROM Alunos
      WHERE Idade < 20 AND Cidade = 'Itu';
```

-  Os predicados $\text{Idade} < 20$ e $\text{Cidade} = \text{'Itu'}$ são de tipo indexável.
-  O predicado $\text{Idade} < 20$ é indexável tanto por CidadeIdade quanto por IdadeCidadeNome .
-  O predicado $\text{Cidade} = \text{'Itu'}$ é indexável tanto por CidadeIdade quanto por IdadeCidadeNome .

Predicados Indexáveis

Predicados Indexáveis – Exemplo

Seja a relação $\text{Alunos} = \{\underline{\text{NUSP}}, \text{Nome}, \text{Idade}, \text{Cidade}\}$ para o qual foram criados os índices:

Index PK: {NUSP}

Index NomeNUSP: {Nome, NUSP}

Index Cidadeldade: {Cidade, Idade}

Index IdadeCidadeNome: {Idade, Cidade, Nome}

- Dada a consulta:

```
SELECT * FROM Alunos
      WHERE Nome LIKE 'José%' AND NUSP>9000000;
```

- ☞ Os predicados `Nome LIKE 'José%'` e `NUSP>9000000` são de tipo indexável.
- ☞ O predicado `Nome LIKE 'José'` é indexavel por `NomeNUSP`.
- ☞ O predicado `NUSP>9000000` é indexavel tanto por `NomeNUSP` quanto por PK.

Predicados Indexáveis

Predicados Indexáveis – Exemplo

Seja a relação $\text{Alunos} = \{\underline{\text{NUSP}}, \text{Nome}, \text{Idade}, \text{Cidade}\}$
para o qual foram criados os índices:

Index PK: {NUSP}



Index NomeNUSP: {Nome, NUSP}

Index Cidadeldade: {Cidade, Idade}

Index IdadeCidadeNome: {Idade, Cidade, Nome}

- Dada a consulta:

```
SELECT * FROM Alunos
      WHERE Nome LIKE '%José' AND NUSP>9000000;
```

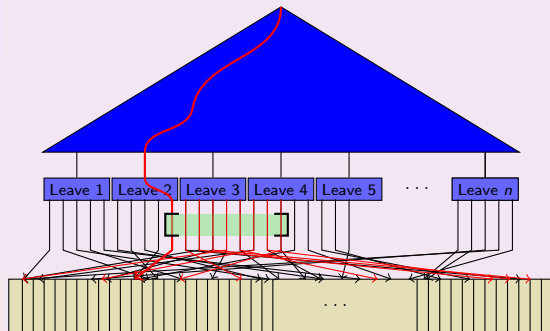
-  O predicado $\text{NUSP} > 9000000$ é de tipo indexável e o predicado Nome LIKE '%José' não é de tipo indexável.
-  O predicado $\text{NUSP} > 9000000$ é indexavel apenas pelo índice por PK.

Predicados de Busca

- Uma consulta Q pode conter predicados para várias finalidades, por exemplo, para estabelecer as chaves estrangeiras entre relações ou para localizar ou restringir tuplas.
- Quando um predicado é usado para localizar ou restringir tuplas e ele é indexável, então o índice correspondente pode ser usado para agilizar a consulta,
- e nesse caso o predicado é chamado **predicado de busca** (*matching predicate*), e o atributo é chamado **atributo de busca**.

Predicados de Busca

- Um índice pode ser usado para localizar ou restringir tuplas da seguinte maneira:



- Quando é feita a busca por um predicado, é feita a **busca indexada** da primeira tupla que atende àquele predicado, e depois continua com uma **busca sequencial** até a última tupla que atende ao predicado.

Predicados de Busca

Procedimento para localizar predicados de busca de um índice idx :

- Um predicado de busca é um predicado indexável por idx ;
- Para cada atributo atr_i que compõe o índice, da esquerda para a direita: *Veja que o loop é na sequência de atributos definida no índice!*
 - 1 Deve existir pelo menos um predicado indexável para o atributo atr_i na consulta Q , e portanto atr_i é um atributo de busca;
 - 2 Se esse é um predicado por faixa, ele é de busca e encerra-se o *loop*;
 - 3 Se o predicado é por lista, e não houve outro predicado por lista antes, ele é de busca. Caso exista outro predicado por lista antes, ele não é de busca e encerra-se o *loop*;
 - 4 Quando não se encontra um predicado com o atributo atr_i , encerra-se a busca.

Predicados de Busca

- Note-se que um predicado pode não ser de busca, mas ainda pode ser um predicado de restrição.
- Por exemplo, pode-se usar um índice de busca que identifica as tuplas a serem recuperadas, e um outro índice que não é de busca, mas com seletividade maior, que pode ser usado para restringir as tuplas que fazem parte da resposta.
- Com isso existem dois conceitos:
 - Um **predicado de restrição** é um predicado que não é de busca mas restringe as tuplas que podem fazer parte da resposta;
 - O uso de predicados de restrição para filtrar as tuplas é uma operação chamada **restrição por predicado**.

Tipos de Predicados – Resumo

- Resumindo, os predicados são classificados de várias maneiras, dependendo de como eles indicam a comparação dos atributos e de como eles são usados para executar as consultas.
- Quanto a maneira como comparam os atributos, existem:
 - **Predicados por Faixa** – usam comparações $\theta \in \{<, \leq, \geq, >\}$,
`LIKE` ou `BETWEEN`;
 - **Predicados por Lista** – usam comparações `IN list` ou expressões `(a1 = c1 OR a2 = c2)`.

Tipos de Predicados – Resumo

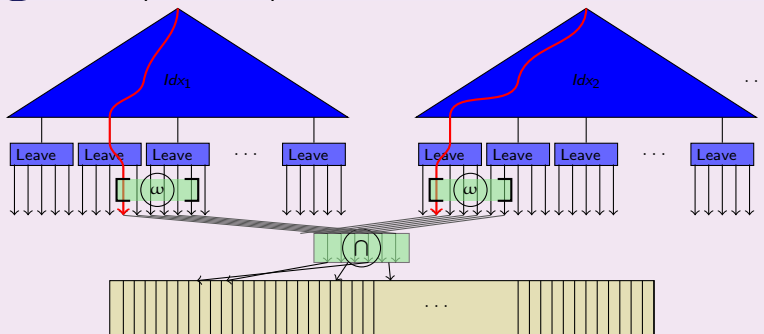
- Quanto à sua capacidade de aproveitar um determinado índice para agilizar as consultas, existem os seguintes tipos de predicados :
 - **Predicados Indexáveis** – O atributo do predicado faz parte do índice, e todos os atributos anteriores a ele no índice são indexáveis;
 - **Predicados de Busca** – O atributo do predicado é indexável pelo índice e o índice pode usar esse atributo para **localizar** as tuplas da resposta;
 - **Predicados de Restrição** – O atributo do predicado é indexável pelo índice e o índice pode prover *ROWIDs* para **restringir** as respostas.

Uso de índices para responder a consultas

- Quando mais de um índice contribui com predicados indexáveis, deve-se avaliar a seletividade composta de todos os predicados associados a cada índice.
- A princípio, apenas um índice pode ser usado para acessar as tuplas no segmento de dados, e nesse caso, o índice com a maior seletividade composta deve ser usado para **localizar** as tuplas.
- Isso é sempre verdade no caso de *clusters* de relações.
- No entanto, é possível usar vários índices para **restringir** as tuplas a serem acessadas.

Uso de índices para responder a consultas

- É possível usar vários índices para **restringir** as tuplas acessadas, assim:
 - 1 Ler a *Rid-list* de cada um dos índices para a memória;
 - 2 Ordenar cada uma das *Rid-list* em memória;
 - 3 executar operações de intersecção ou união das *Rid-lists* (dependendo dos predicados estarem ligados por AND ou OR);
 - 4 acessar apenas as tuplas resultantes.



Uso de índices para responder a consultas

- Assim, o cálculo da seletividade dos predicados da consulta é fundamental para a escolha do melhor método de acesso aos dados;
- e durante esse cálculo, identificam-se os índices que podem potencialmente contribuir para agilizar o acesso.
- No entanto, como se verá na sequência, o acesso físico depende de se levar em consideração também as formas de **Acesso Físico ao Disco**.

Coleta de estatísticas em SQL

- O DBA utiliza o comando `ANALYZE STATISTICS` para solicitar a análise dos dados armazenados e coletar as estatísticas.
- As estatísticas são armazenadas na Metabase, e são usadas nos comandos subsequentes para otimizar a construção dos planos de acesso.

Comando ANALYZE STATISTICS

Postgres

ANALYZE STATISTICS – Postgres

```
ANALYZE [VERBOSE] [tabela [(atributo [, ...])]]
```

- Postgres escreve as estatísticas na meta-tabela `pg_statistic`
- Se nenhuma tabela for especificada, as estatísticas são geradas para toda a base.

Comando ANALYZE STATISTICS

Oracle

ANALYZE STATISTICS – Oracle

```
ANALYZE TABLE <tabela>
  {COMPUTE [SYSTEM] STATISTICS |
  DELETE STATISTICS |
  ESTIMATE [SYSTEM] STATISTICS
    [SAMPLE <numero> {ROWS | PERCENT}]}
  [FOR TABLE | ALL INDEXES |
    ALL [INDEXED] COLUMNS [SIZE <faixas>]
    COLUMNS (<atributo> [SIZE <faixas>], )
  [INTO <tabela>]
```

- Desde a versão 9, Oracle tem recomendado que ANALYZE STATISTICS seja usado apenas para análise dos dados pelo DBA. Para usar no otimizador, deve-se usar a ferramenta DBMS_STATS, que roda concorrente.
- SIZE <faixas> indica o tamanho máximo do histograma a ser usado.

Arquitetura de SGBD Relacionais — Previsão de Seletividade —

Caetano Traina Jr.

Grupo de Bases de Dados e Imagens
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo - São Carlos
caetano@icmc.usp.br

20 de abril de 2013
São Carlos, SP - Brasil

FIM