

Sistemas de Banco de Dados

Fundamentos em Bancos de Dados Relacionais

Wladmir Cardoso Brandão

www.wladmirbrandao.com

Material distribuído sob licença CC BY-NC-ND 4.0

Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International

PROJETO FÍSICO



Criação de estrutura apropriada para armazenamento de dados com foco em desempenho na execução de consultas e transações

FATOR DE IMPACTO

- ▶ **CARACTERÍSTICA DA CONSULTA**
- ▶ Frequência de execução
- ▶ Tempo de execução
- ▶ Exclusividade em campo

DECISÃO

- ▶ Recuperação de dados
 - ▶ Que arquivos são acessados?
 - ▶ Que campos estão especificados?
 - ▶ JUNÇÃO e SELEÇÃO → indexação
 - ▶ Condição em JUNÇÃO e SELEÇÃO?
 - ▶ $=$ → multiplicidade de índices
 - ▶ \neq → limita uso de índice
 - ▶ $<\leq\geq>$ → limitação índice *hash*



Criação de estrutura apropriada para armazenamento de dados com foco em desempenho na execução de consultas e transações

FATOR DE IMPACTO

- ▶ **CARACTERÍSTICA DA CONSULTA**
- ▶ Frequência de execução
- ▶ Tempo de execução
- ▶ Exclusividade em campo

DECISÃO

- ▶ Atualização de dados
 - ▶ Que arquivos são atualizados?
 - ▶ Que operações são realizadas?
 - ▶ Que campos estão especificados?
 - ▶ SELEÇÃO → considerar indexação
 - ▶ MODIFICAÇÃO → evitar indexação
- ▶ Condição em SELEÇÃO?
 - ▶ = → multiplicidade de índices
 - ▶ ≠ → limita uso de índice
 - ▶ < ≤ ≥ > → limitação índice *hash*



Criação de estrutura apropriada para armazenamento de dados com foco em desempenho na execução de consultas e transações

FATOR DE IMPACTO

- ▶ Característica da consulta
- ▶ **FREQUÊNCIA DE EXECUÇÃO**
- ▶ Tempo de execução
- ▶ Exclusividade em campo

DECISÃO

- ▶ Execuções por unidade de tempo?
 - ▶ **FREQUENTE** → considerar indexação para recuperação frequente e evitar indexação para atualização frequente
 - ▶ **PARETO** → 80% do processamento é consumido por 20% das consultas e transações
 - ▶ Top 20% → considerar indexação para as de recuperação e evitar indexação para as de atualização



Criação de estrutura apropriada para armazenamento de dados com foco em desempenho na execução de consultas e transações

FATOR DE IMPACTO

- ▶ Característica da consulta
- ▶ Frequência de execução
- ▶ **TEMPO DE EXECUÇÃO**
- ▶ Exclusividade em campo

DECISÃO

- ▶ Tempo médio e máximo esperado para execução?
 - ▶ Que consultas e transações tem forte restrição de tempo de execução?
 - ▶ CRÍTICAS → considerar indexação para recuperação e evitar indexação para atualização



Criação de estrutura apropriada para armazenamento de dados com foco em desempenho na execução de consultas e transações

FATOR DE IMPACTO

- ▶ Característica da consulta
- ▶ Frequência de execução
- ▶ Tempo de execução
- ▶ **EXCLUSIVIDADE EM CAMPO**

DECISÃO

- ▶ Que campos são exclusivos?
 - ▶ Campos exclusivos são usados frequentemente em **JUNÇÃO** e **SELEÇÃO**
 - ▶ Considerar indexação para campos exclusivos



Muitas decisões de projeto físico envolvem indexação

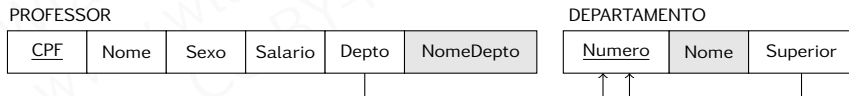
```
CREATE [UNIQUE] INDEX <nome>  
ON <tabela> ( <coluna> [<ORDEM>] {, <coluna> [<ORDEM>]} )  
[CLUSTER];
```

- ▶ UNIQUE → campo de indexação será exclusivo
- ▶ CLUSTER → índice será um arquivo ordenado pelo campo de indexação
- ▶ ORDEM → forma de ordenação do campo de indexação
 - ▶ ASC → ordenação ascendente
 - ▶ DESC → ordenação descendente



DESNORMALIZAÇÃO → modificação no projeto lógico para se obter mais eficiência no processamento de consultas e transações

- ▶ Duplicação de atributos → inclusão de atributos de uma tabela em outra
 - ▶ Evita operações de JUNÇÃO entre as tabelas
 - ▶ Introduce **redundância** em tabelas
 - ▶ Exemplo: introduzir o atributo *Nome* da tabela DEPARTAMENTO na tabela PROFESSOR evita a necessidade de JUNÇÃO entre as tabelas se a consulta por *Nome* de DEPARTAMENTO e *Nome* de PROFESSOR for frequentemente realizada





Todo projeto precisa de ajustes ao ser executado

SINTONIA (TUNING) → processo de ajuste contínuo do projeto físico

- ▶ Monitora e revisa decisões de projeto
 - ▶ OBJETIVO → redução do tempo de processamento de consultas e transações
 - ▶ MÉTRICAS → conjunto de medidas usadas no monitoramento
 - ▶ PROCESSAMENTO → tempo de otimização e execução de consultas e transações
 - ▶ ARMAZENAMENTO → espaços ocupados por *pools* de *buffer*, tabelas (*tablespaces*) e índices (*indexspaces*)
 - ▶ I/O → # paginações em disco por unidade de tempo
 - ▶ CONCORRÊNCIA → taxa de vazão (*throughput*) de transações, de emissão de comandos de bloqueio, e de registro em *log*
 - ▶ ÍNDICE → # níveis, # nós folha não contíguos



DBAs monitoram métricas para realizar ajustes e evitar problemas:

- ▶ DESPERDÍCIO → tamanho de *buffers* inadequados
- ▶ SOBRECARGA → *logging* e *dumping* desnecessários
- ▶ AUMENTO DE CONCORRÊNCIA → disputa excessiva por bloqueios
- ▶ INEFICIÊNCIA → alocação inadequada de discos, *buffers* e processos

Tais ajustes podem ocorrer de diferentes formas

- ▶ SINTONIA DE ÍNDICE → criação, remoção e reorganização de índices
- ▶ SINTONIA DE PROJETO → alterações no projeto lógico
- ▶ SINTONIA DE CONSULTA → reescrita de consultas



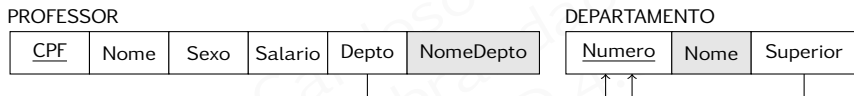
Avaliação dos requisitos de projeto físico para reorganizar arquivos e índices

- ▶ CRIAÇÃO → consultas e transações podem estar demorando a serem executadas por ausência de índice
- ▶ REMOÇÃO → índices podem estar sendo pouco utilizados
- ▶ REORGANIZAÇÃO → índices podem estar sendo muito atualizados
 - ▶ EXCLUSÃO → blocos de índice com espaço desperdiçado
 - ▶ INCLUSÃO → *overflow* excessivo em índice agrupado
- ▶ Opções de indexação variam em soluções de SGBD comerciais
 - ▶ SYBASE → índices de agrupamento esparsos em B+ TREES
 - ▶ INGRESS → índices de agrupamento ISAM esparsos ou B+ TREES densos
 - ▶ ORACLE → índices de agrupamento densos

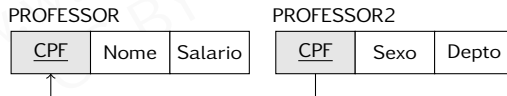


Avaliação dos requisitos de projeto físico para modificação dos projetos conceitual e lógico

- DUPLICAÇÃO DE ATRIBUTOS → inclusão de atributos de uma tabela em outra



- PARTICIONAMENTO VERTICAL → divisão de atributos de uma tabela em múltiplas tabelas com mesma chave primária em relacionamento 1 : 1





Avaliação dos requisitos de projeto físico para modificação dos projetos conceitual e lógico

- ▶ PARTICIONAMENTO HORIZONTAL → distribuição de tuplas de uma tabela em múltiplas tabelas com os mesmos atributos

PROFESSOR

<u>CPF</u>	Nome	Sexo	Salario	Departamento
12345678900	Roberto Machado	M	1200.00	1
12345678901	Manuela Costa	F	2700.00	3

PROFESSOR TOP

<u>CPF</u>	Nome	Sexo	Salario	Departamento
21345678900	Carlos A. Martins	M	3200.00	1
32145678900	Ana Maria Freitas	F	7500.00	2



Avaliação dos requisitos de projeto físico para reescrever consultas

- ▶ INDÍCIOS → sinais de que consultas precisam ser reescritas
 - ▶ PLANO DE EXECUÇÃO → índices relevantes não estão sendo usados
 - ▶ PAGINAÇÃO → emissão de muitas solicitações de I/O
- ▶ CASOS TÍPICOS → situações que demandam reescrita de consulta
 - ▶ PARSING → ordem aleatória de tabelas no FROM e operações no WHERE
 - ▶ COMPARAÇÃO → **NULL**, *substring* e **campos de domínios diferentes**

```
SELECT  A.CPF, B.Nome
FROM    PROFESSOR A, DEPARTAMENTO B
WHERE   A.Depto IS NOT NULL
        AND B.Nome LIKE '%TI%'
        AND A.Salario = B.Numero;
```



Avaliação dos requisitos de projeto físico para reescrever consultas

► CASOS TÍPICOS → situações que demandam reescrita de consulta

► CONSULTAS ANINHADAS → operadores ALL, ANY, SOME, IN e EXISTS

```
SELECT CPF, Nome FROM PROFESSOR  
WHERE Depto IN (SELECT Numero FROM DEPARTAMENTO);
```

```
SELECT A.CPF, A.Nome FROM PROFESSOR A  
WHERE EXISTS (SELECT * FROM DEPARTAMENTO B  
              WHERE A.Depto = B.Numero);
```

► DEDUPLICAÇÃO → operador DISTINCT

```
SELECT DISTINCT Nome FROM PROFESSOR;
```




Avaliação dos requisitos de projeto físico para reescrever consultas

- ▶ CASOS TÍPICOS → situações que demandam reescrita de consulta
 - ▶ CONDIÇÃO DISJUNTIVA → operador OR

```
SELECT CPF, Nome FROM PROFESSOR  
WHERE Sexo = 'M' OR Salario > 2000,00;
```

```
SELECT CPF, Nome FROM PROFESSOR  
WHERE Sexo = 'M'
```

UNION

```
SELECT CPF, Nome FROM PROFESSOR  
WHERE Salario > 2000,00;
```



Avaliação dos requisitos de projeto físico para reescrever consultas

- ▶ CASOS TÍPICOS → situações que demandam reescrita de consulta
 - ▶ CONDIÇÃO COMPLEXA → operadores AND e OR

```
SELECT CPF, Nome FROM PROFESSOR
WHERE Depto = 1
      AND ((Salario BETWEEN 1000,00 AND 2000,00) OR
           (Salario BETWEEN 5000,00 AND 7000,00));
```

```
SELECT CPF, Nome FROM PROFESSOR
WHERE Depto = 1 AND (Salario BETWEEN 1000,00 AND 2000,00)
UNION
SELECT CPF, Nome FROM PROFESSOR
WHERE Depto = 1 AND (Salario BETWEEN 5000,00 AND 7000,00);
```



Avaliação dos requisitos de projeto físico para reescrever consultas

- ▶ CASOS TÍPICOS → situações que demandam reescrita de consulta
 - ▶ CONSULTA COMPLEXA → subconsultas

```
SELECT A.CPF, A.Nome FROM PROFESSOR A
WHERE A.Salario > (SELECT AVG(B.Salario) FROM PROFESSOR B
WHERE A.Depto = B.Depto);
```

```
SELECT      Depto, AVG(Salario) AS Media INTO TEMP
FROM        PROFESSOR
GROUP BY    Depto;
```

```
SELECT      A.CPF, A.Nome FROM PROFESSOR A, TEMP B
WHERE       A.Salario > B.Media
AND         A.Depto = B.Depto;
```



- [1] Elmasri, Ramez; Navathe, Sham. *Fundamentals of Database Systems*. 7ed. Pearson, 2016.
- [2] Silberschatz, Abraham; Korth, Henry F.; Sudarshan, S. *Database System Concepts*. 6ed. McGraw-Hill, 2011.
- [3] Date, Christopher J. *An Introduction to Database Systems*. 8ed. Pearson, 2004.