

Banco de Dados

MODELO FÍSICO

DDL – ESTRUTURA DE INDEXAÇÃO DAS TABELAS

GLAUCO KISS LEME

Agenda

2

- ▶ O que são índices?
- ▶ Qual o objetivo do Índice?
- ▶ Tipos de Índices
 - ▶ Cluster Index;
 - ▶ Non-Cluster Index;
 - ▶ Cover Index;
 - ▶ Tabelas de Catalogo;



O que são Índices

3

▶ O que são Índices?

- ▶ Índices são estruturas que possuem algoritmos otimizados para acessar dados em um banco de dados.

▶ Qual o objetivo do Índice?

- ▶ Acelerar as consultas feitas no Banco de Dados.



- ▶ **Qual é o trade-off?**

- ▶ Custo de Manutenção X Custo de Pesquisa
- ▶ A escolha dos índices deve ser feita de maneira coerente, de acordo com as necessidades de negócio e da estrutura do projeto de Banco de Dados.

- ▶ **O que acontece se eu não utilizar Índices?**

- ▶ Ao não utilizar índices, o SGBD precisa realizar uma operação conhecida como **Table Scan**, para localizar os dados solicitados.
- ▶ **Table Scan** é uma leitura seqüencial de todos os registros da tabela (linha a linha), o que implica muito mais operações de I/O no disco.

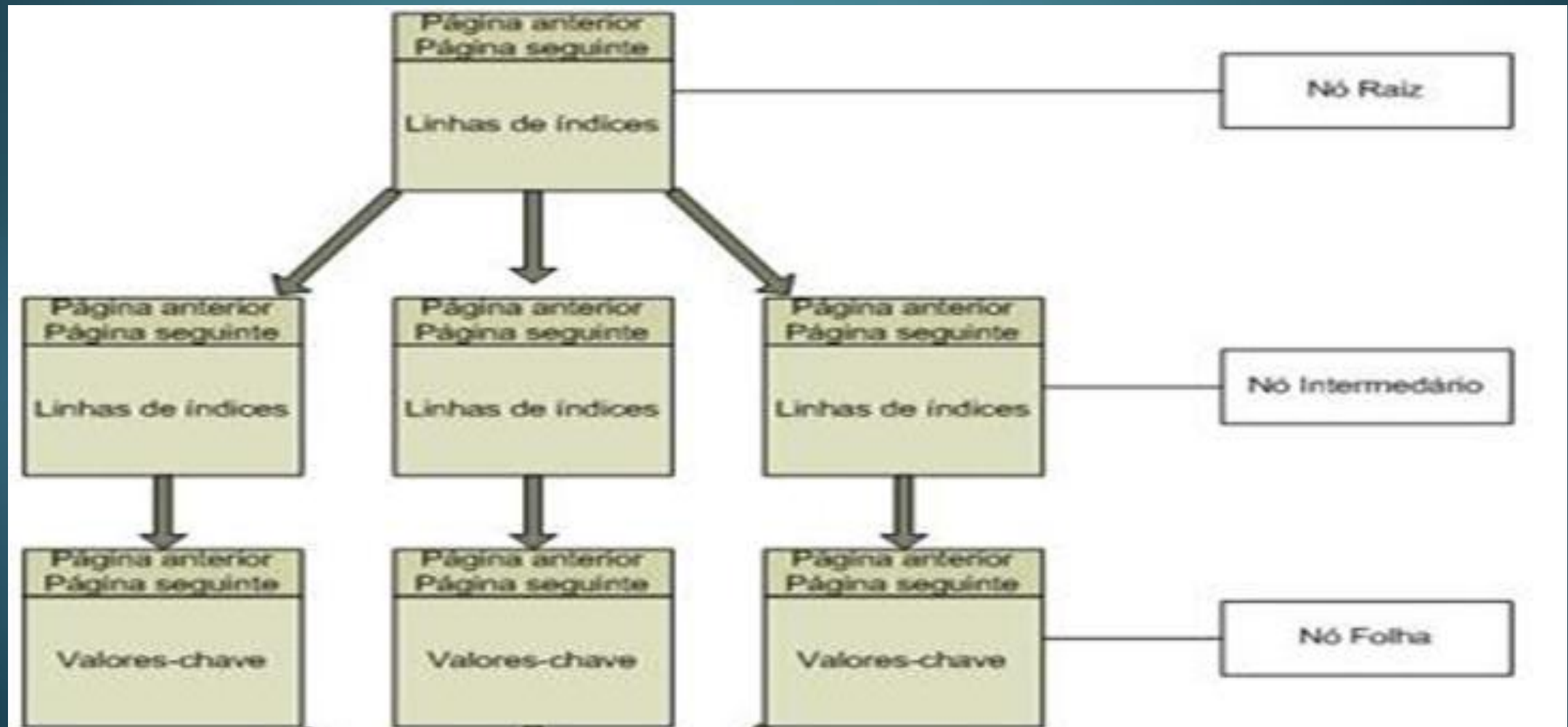
- ▶ Devemos ao máximo evitar os *table scans* em um banco de dados SQL Server. Se partirmos do princípio que nosso banco de dados passou por um processo de normalização dos dados, dificilmente teremos uma tabela sem pelo menos um índice primário (PK).
- ▶ Mas mesmo assim, o SQL Server, poderá achar que é “menos custoso”, acessar o dado de uma tabela, realizando uma varredura física, do que ter que acessar a página de índices para recuperar a informação. Isso poderá acontecer em tabelas com pequenas quantidades de dados.

Índices

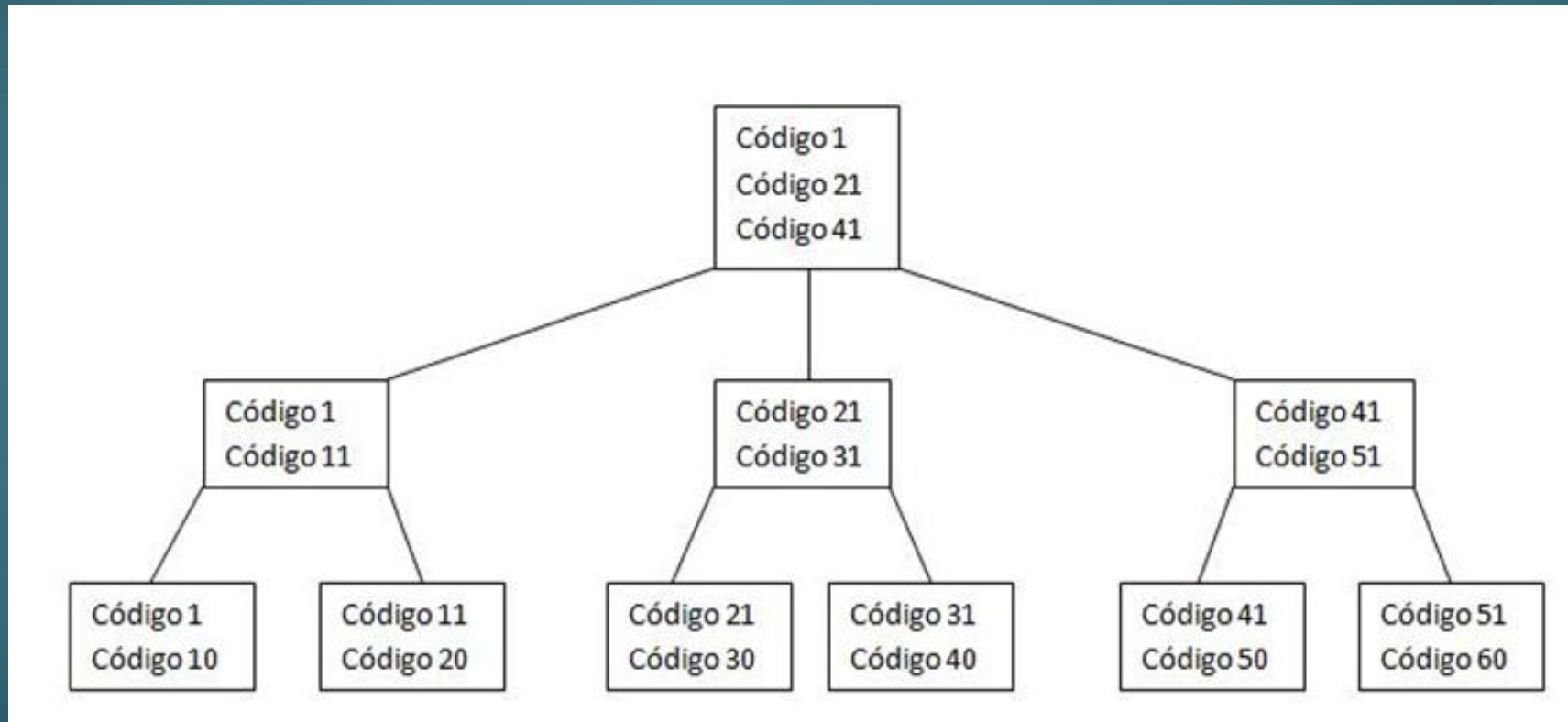
- ▶ Todo índice no SQL Server é organizado em uma B-tree.
- ▶ É uma estrutura de dados em formato de árvore onde o B significa balanceada.

Índices

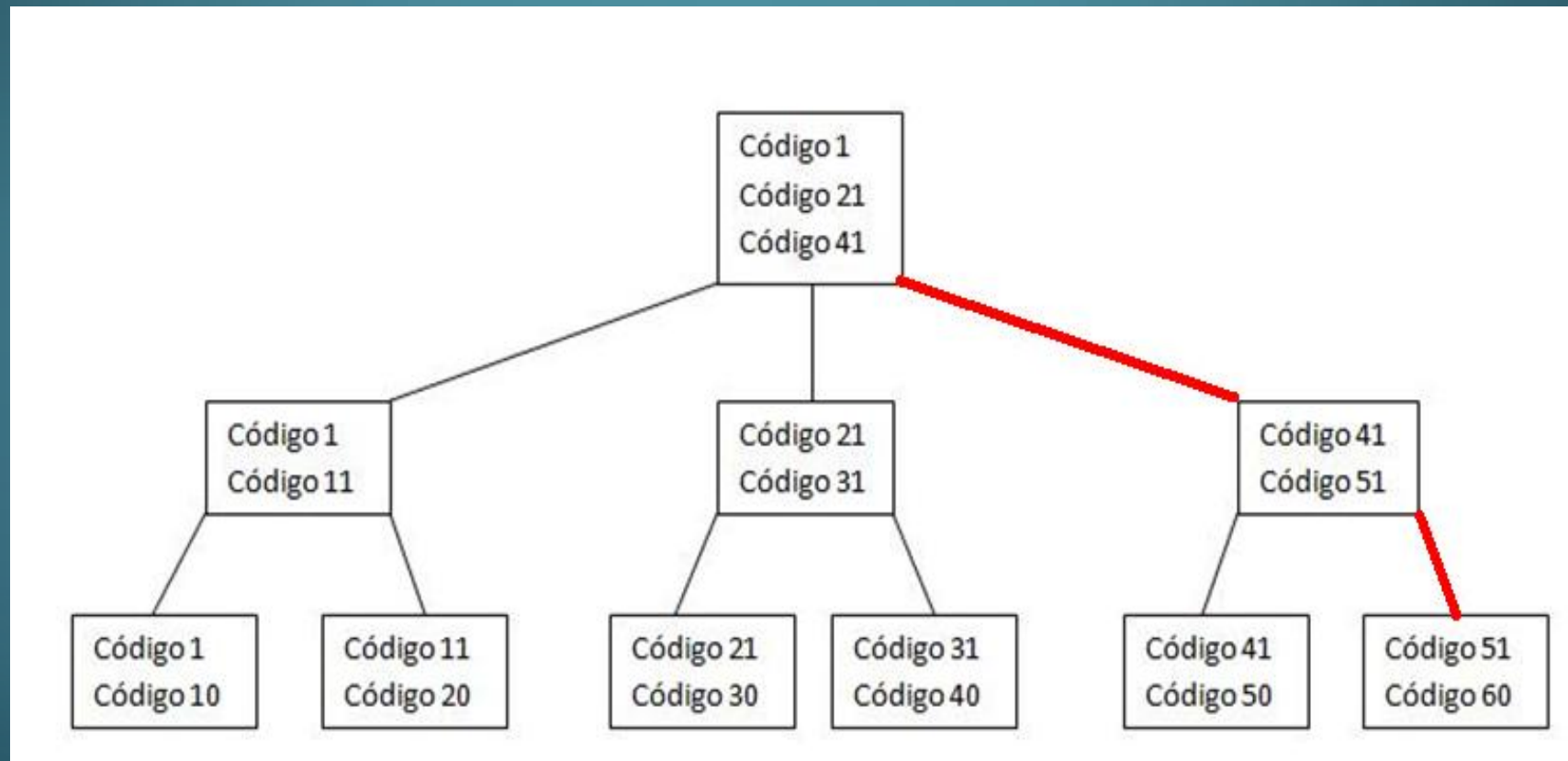
7



- ▶ Encontre o registro de código 57



- ▶ Encontre o registro de código 57



Tipos de Índices no SQL Server 2008+

10

- ▶ Índice Cluster;
- ▶ Índice Non-Cluster;
- ▶ Cover Index;

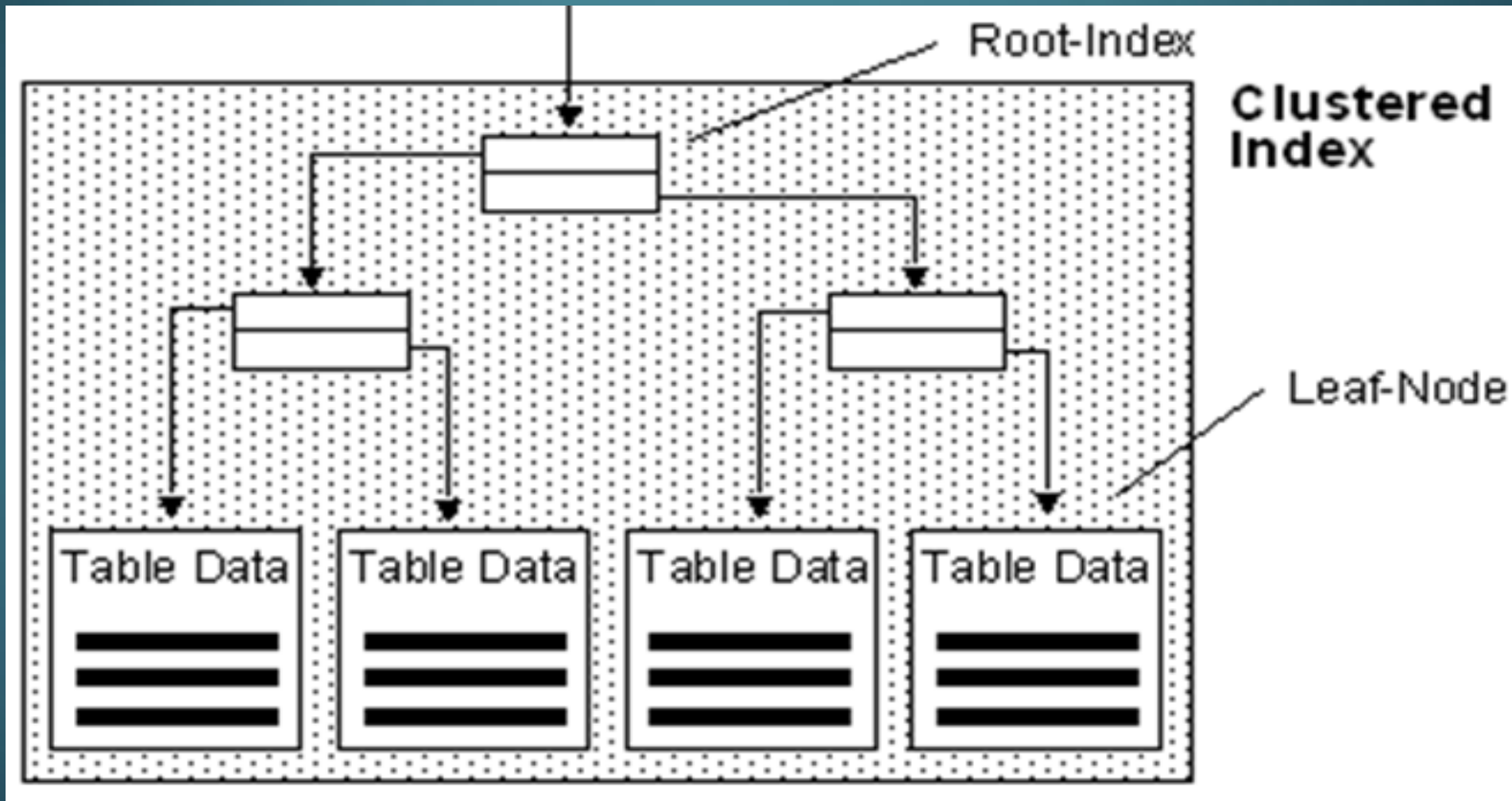
Índice Cluster

11

- ▶ É um índice gerado na própria estrutura de armazenamento dos dados, esse tipo de índice, fará com que os dados da tabela fiquem organizados fisicamente na sequencia.
- ▶ É possível definir somente um índice clusterizado por tabela, pois a mesma só pode ser ordenada de uma única maneira.
- ▶ O SQL Server cria um índice cluster automaticamente quando criamos uma primary key.
- ▶ Tabelas que não possuem um Índice Cluster são chamadas de **Heap**.

Índice Cluster

12



Índice Cluster – Sintaxe Criação

13

► Sintaxe:

```
CREATE [UNIQUE] CLUSTERED INDEX nome_index  
ON nome_tabela(nome_campo)
```

► Exemplos:

```
-- Criando um índice Cluster e único para o campo IDCliente da tabela  
  Cliente.
```

```
CREATE UNIQUE CLUSTERED INDEX IDX_IDCliente  
ON Cliente(IDCliente)
```

Índice Cluster – Sintaxe Exclusão

14

► Sintaxe

```
DROP INDEX nome_tabela.nome_index
```

Ou

```
DROP INDEX nome_index ON nome_tabela
```

► Exemplos:

```
-- Excluindo o índice IDX_NomeCliente
```

```
DROP INDEX Cliente.IDX_IDCliente
```

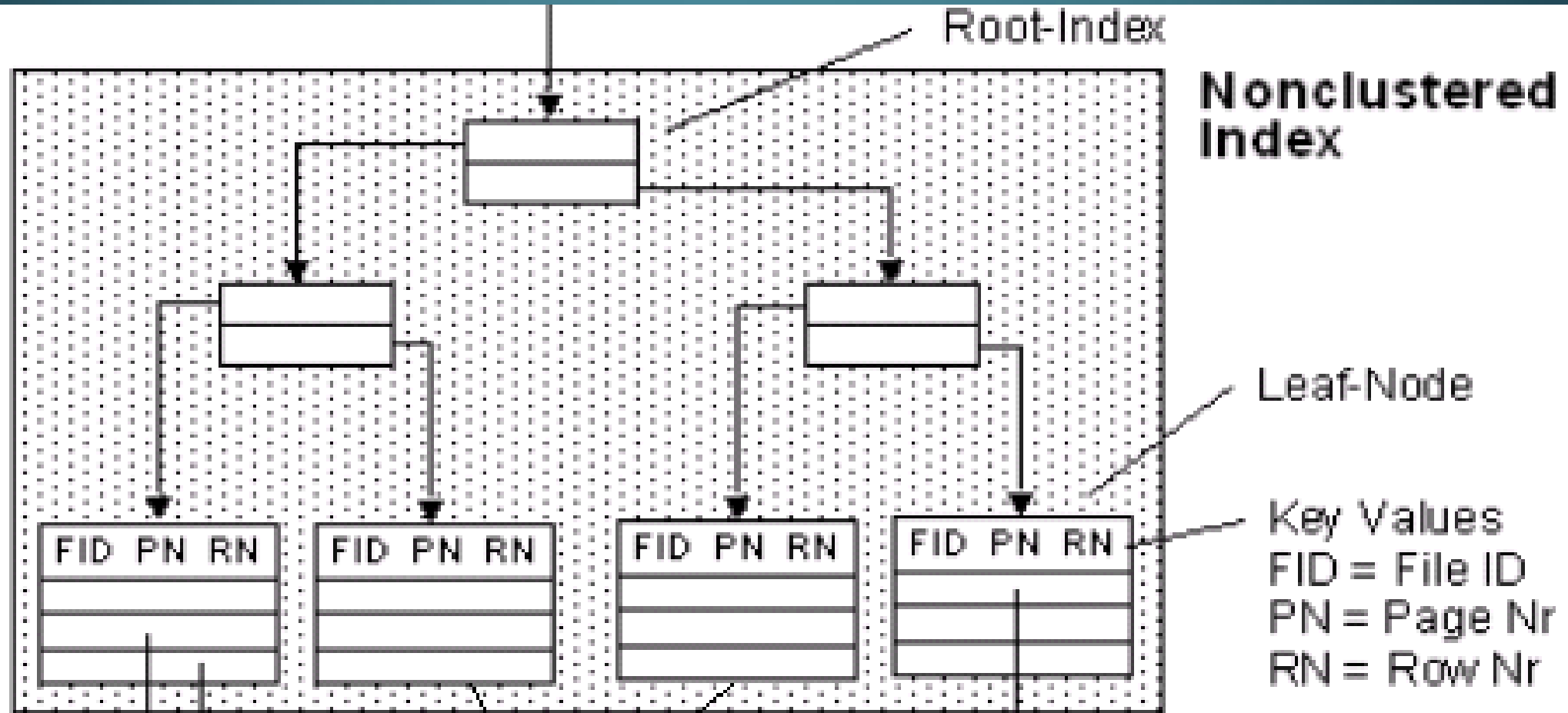
Índice Non-Cluster

15

- ▶ Estrutura semelhante ao cluster, com exceção do nível folha possuir um bookmark para os demais dados da tabela.
- ▶ Podem existir até 254 índices Não Cluster por tabela.

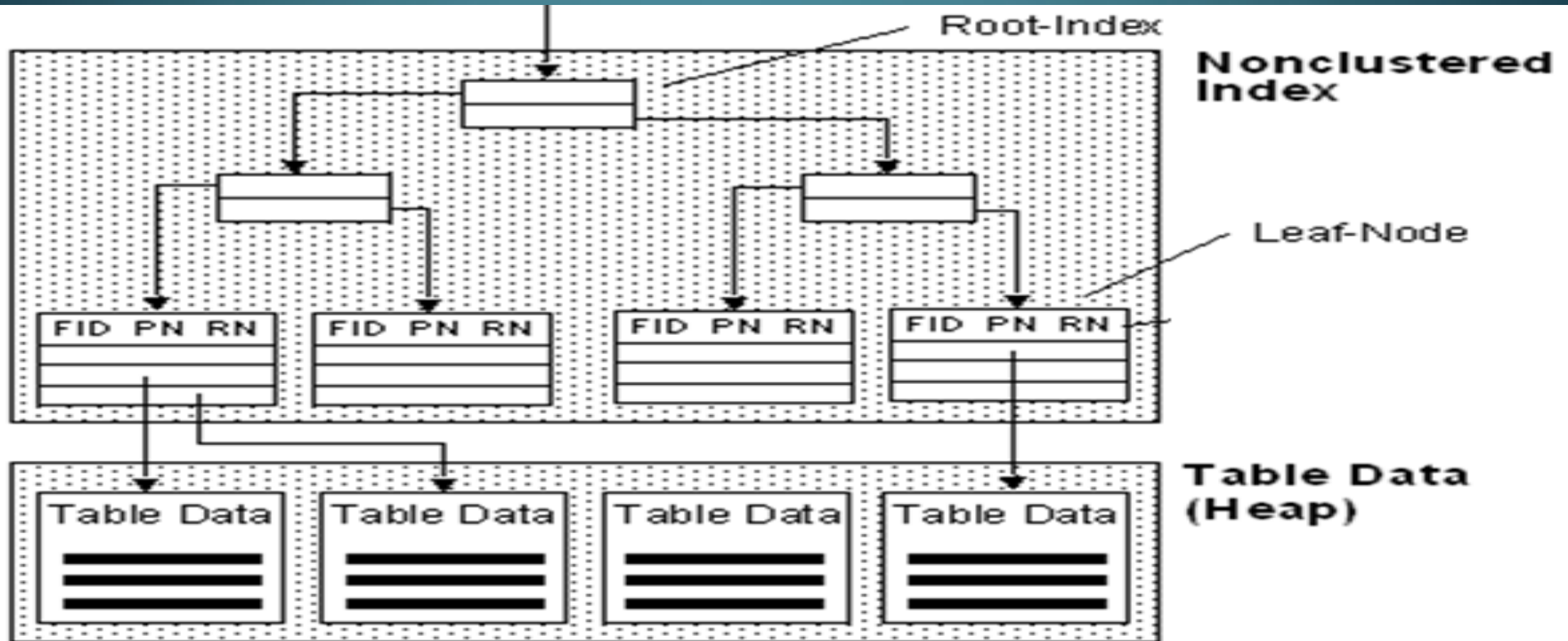
Índice Non-Cluster

16



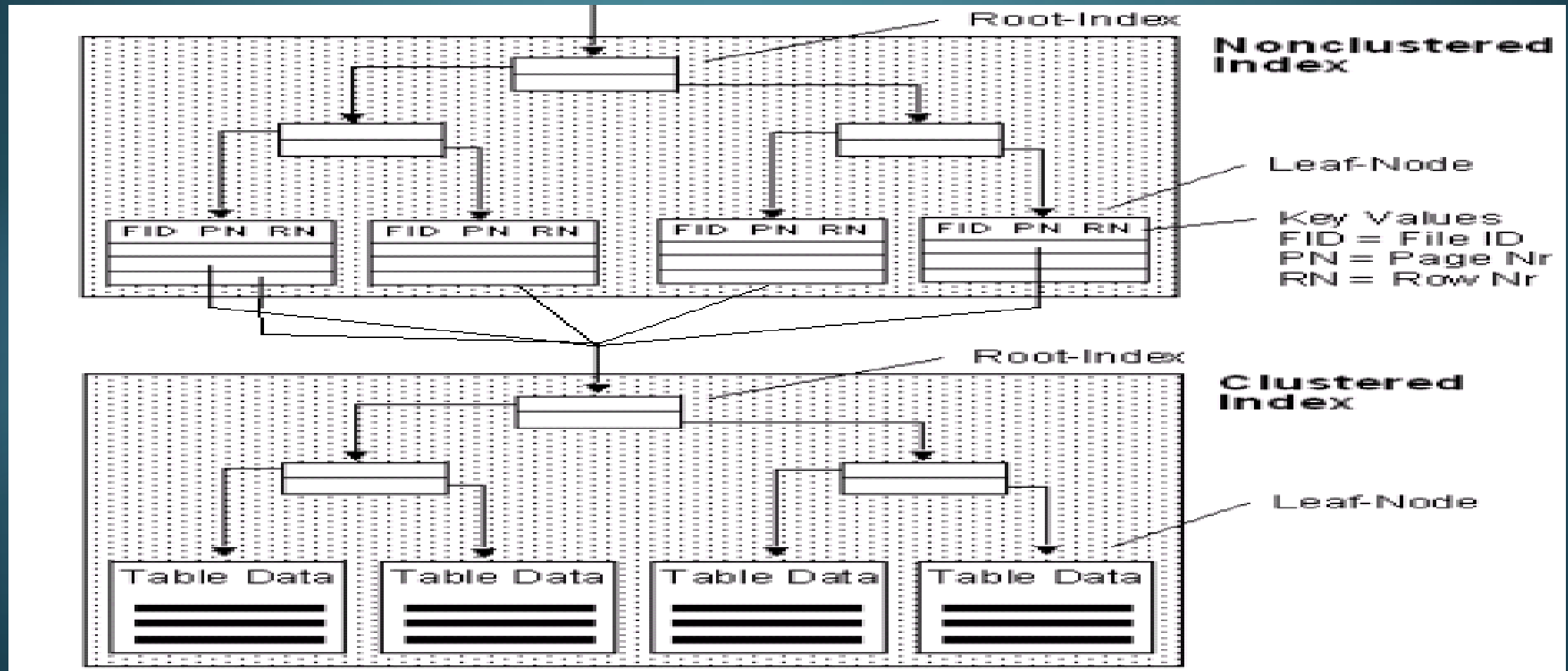
Índice Non-Cluster acessando dados que não estão no índice (Não existe um Cluster Index)

17



Índice Non-Cluster acessando dados que não estão no índice (existe um Cluster Index)

18



► Sintaxe:

```
CREATE [UNIQUE] NONCLUSTERED INDEX nome_index  
ON nome_tabela(nome_campo)
```

► Exemplos:

```
-- Criando um índice Não-Cluster que permite repetições para o campo  
NomeCliente da tabela Cliente.
```

```
CREATE NONCLUSTERED INDEX IDX_NomeCliente  
ON Cliente(NomeCliente)
```

► Sintaxe:

```
DROP INDEX nome_tabela.nome_index
```

Ou

```
DROP INDEX nome_index ON nome_tabela
```

► Exemplos:

```
-- Excluindo o índice IDX_NomeCliente
```

```
DROP INDEX Cliente.IDX_NomeCliente
```

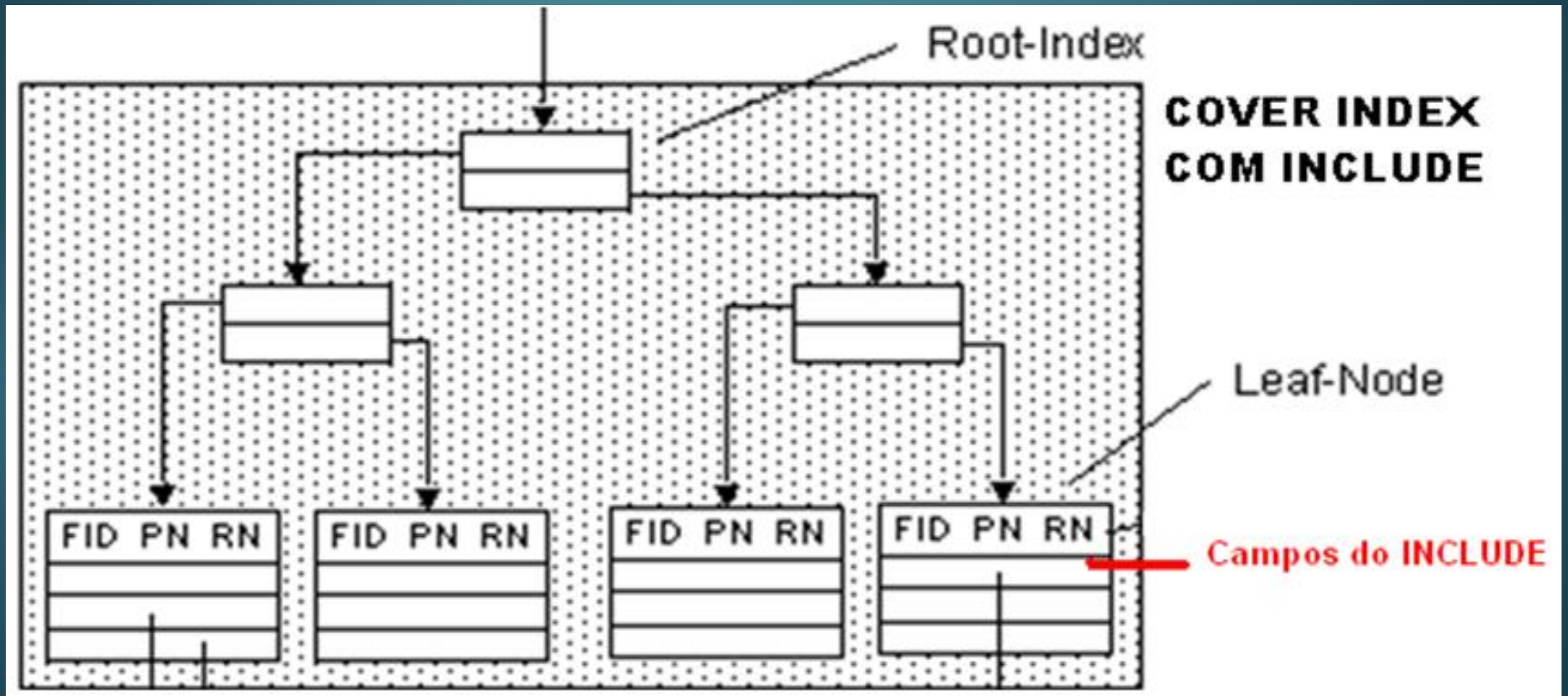
Cover Índices

21

- ▶ São índices que “cobrem” a consulta, evitando que seja feito o bookmark.
- ▶ Para cobrir a consulta, **todas as colunas necessárias devem estar presentes no índice não cluster.**
- ▶ A partir do SQL 2005 existe a opção INCLUDE.
- ▶ Esta opção permite incluir informações adicionais (campos) nos nós folha.

Cover Índices

22



Cover Index - Sintaxe de Criação

23

```
CREATE [UNIQUE] NONCLUSTERED INDEX nome_index  
ON nome_tabela(nome_campo)  
INCLUDE nome_campo1, nome_campo2, ...)
```

► Exemplos:

-- Criando um índice Não-Cluster, para o campo NomeCliente da tabela Cliente, incluindo também os campos DataNascimento e Endereço.

```
CREATE NONCLUSTERED INDEX IDX_NomeCliente  
ON Cliente(NomeCliente)  
INCLUDE DataNascimento, Endereco)
```

Cover Index - Sintaxe de Criação

24

► Sintaxe:

```
DROP INDEX nome_tabela.nome_index
```

Ou

```
DROP INDEX nome_index ON nome_tabela
```

► Exemplos:

```
-- Excluindo o índice IDX_NomeCliente
```

```
DROP INDEX Cliente.IDX_NomeCliente
```

Tabelas de Catalogo (Gerenciamento)

25

- ▶ O MS SQL Server disponibiliza alguns recursos para visualização e gerenciamento dos índices, no MS SQL 2005 ou superior um dos comandos mais usados são:

```
-- Exibir os índices da tabela.
```

```
SELECT * FROM sys.indexes i
```

```
WHERE i.[object_id] = OBJECT_ID('Nome_da_Tabela')
```

```
-- Exibir os índices da tabela.
```

```
EXEC sys.sp_helpindex 'Nome_da_Tabela'
```

```
-- Obter as páginas do índice.
```

```
SELECT s.[name], *
```

```
FROM sys.allocation_units au
```

```
INNER JOIN sys.partitions p ON au.container_id = p.partition_id
```

```
INNER JOIN sys.indexes s ON s.index_id= p.index_id AND s.[object_id] = p.[object_id]
```

```
WHERE p.[object_id] = OBJECT_ID 'Nome_da_Tabela'
```

- ▶ Índices devem ser criados em colunas usadas para pesquisa, ordenação ou agrupamento, ou seja, nas que aparecem na cláusula WHERE, em joins, ORDER BY ou GROUP BY.
- ▶ Índices funcionam melhor em colunas que contenham um alto número de valores distintos
- ▶ Mantenha, sempre que possível, chaves primárias pequenas (os SGBDs criam um índice para cada chave primária).
- ▶ Não crie índices em excesso. Índices devem ser atualizados e reorganizados quando o conteúdo de uma tabela é modificado. Logo, quanto mais índices, mais demorada será a atualização e a ordenação.

Recomendações

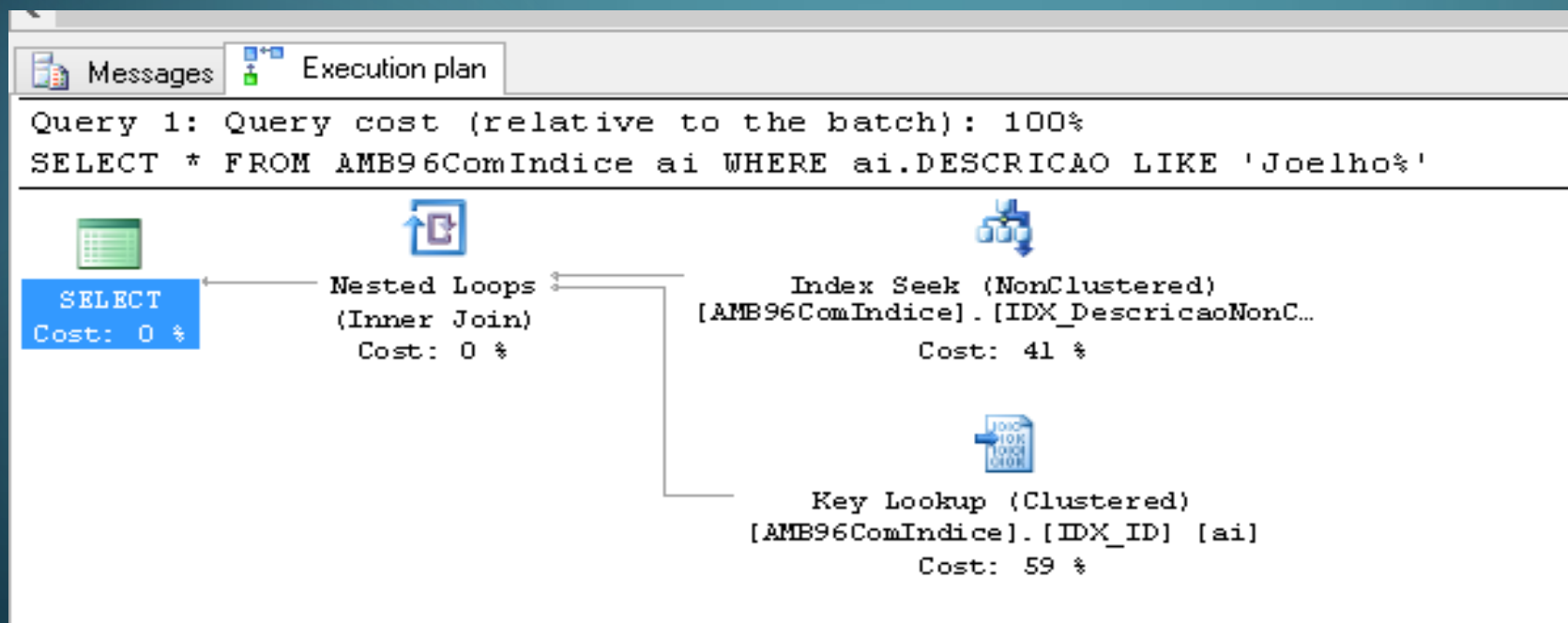
27

- ▶ **Crie índices com valores pequenos. Use tipos de dados o menor possível. Por exemplo, não use uma coluna BIGINT se MEDIUMINT suporta os dados que serão armazenados.**
 - ▶ Não use CHAR(100) se nenhum dos valores armazenados ultrapassa 25 caracteres.
- ▶ **Valores pequenos melhoram o processamento de índices de muitas maneiras:**
 - ▶ Podem ser comparados mais rapidamente
 - ▶ Ocupam menos espaço de disco nos arquivos de índices.

Recomendações

28

- Utilize o plano de execução como forma de analisar a performance da sua consulta.



Vamos por a mão na massa...

29



Mão na massa...

30

1. Apresente uma vantagem e uma desvantagem na utilização dos Índices.
2. Descreva a diferenças entre os tipos de índices apresentados.
3. Considere que o seu Sistema está apresentando um tempo de execução lento em pesquisas na tabela Cidade onde o parâmetro de entrada seja o Nome da Cidade. Utilizando as técnicas vistas durante o curso implemente uma solução para resolver este problema. Faça isso no banco Loja
4. Considere que a empresa está passando por uma Auditoria e que durante alguns dias será necessário imprimir varias vezes um relatório de controle do estoque . Este relatório possui como parâmetro de entrada : o Nome do Produto, e os dados retornados são: Nome do Produto, Qtd em Estoque, e o Situação. Mas diante da grande quantidade de registros desta tabela a impressão deste relatório está muito lenta. Implemente uma solução para resolver este problema. Faça isso no banco Loja.
5. Seu cliente solicitou uma consulta no cadastro de Clientes em que os parâmetros de entrada sejam: o Nome do Cliente , o Número do CPF ou o Número do Telefone, analise este cenário e descreva uma solução que tenha o tempo retorno mais eficiente e implemente no banco Loja.
6. [OPCIONAL] Execute o arquivo AMB96 - SQL.sql e faça as análises do plano de execução confirme feito durante a aula sobre Índices.