

cin.ufpe.br



**Centro
de Informática**
U • F • P • E

Bancos de Dados Distribuídos e Móveis
Aula 01

Bernadette Farias Lóscio
bfl@cin.ufpe.br



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Projeto de Distribuição de Banco de Dados

- Como os dados serão distribuídos ao longo da rede?
- Que unidades lógicas do banco de dados serão distribuídas?
- Fragmentação de dados
 - Horizontal
 - Vertical
- Replicação
 - Total
 - Parcial

Fragmentação Horizontal

- Particiona a relação em conjuntos de tuplas
- Cada fragmento horizontal é especificado a partir de uma condição em um ou mais atributos da relação
- Um fragmento horizontal pode ser especificado por: $\sigma_{Ci}(R)$



Fragmentação Horizontal - Exemplo

Projeto 1: Projetos com orçamento ≥ 100.000
 Projeto 2: Projetos com orçamento < 100.000 } Fragmentos

Projeto

<u>Cod_proj</u>	Orçamento	Cidade
P01	100.000	Fortaleza
P02	120.000	Recife
P03	80.000	Natal
P04	100.000	Natal

Projeto 1

<u>Cod_proj</u>	Orçamento	Cidade
P01	100.000	Fortaleza
P02	120.000	Recife
P04	100.000	Natal

$\sigma_{\text{Orçamento} \geq 100.000}(\text{Projeto})$

Projeto 2

<u>Cod_proj</u>	Orçamento	Cidade
P03	80.000	Natal

$\sigma_{\text{Orçamento} < 100.000}(\text{Projeto})$



Fragmentação Horizontal

- Um conj. de fragmentos horizontais cujas condições C_1, C_2, \dots, C_n incluem todas as tuplas de R, i.e., toda tupla em R que satisfaz (C_1 ou C_2 ou...ou C_n), é chamado de fragmentação horizontal completa
- Em muitos casos uma fragmentação horizontal também é disjunta (nenhuma tupla em R satisfaz (C_i e C_j) para qualquer $i \neq j$)
- Para reconstruir R a partir de uma fragmentação horizontal completa deve ser aplicada a operação de UNION nos fragmentos horizontais



Fragmentação Horizontal - Exemplo

Projeto

Cod_proj	Orçamento	Cidade
P01	100.000	Fortaleza
P02	120.000	Recife
P03	80.000	Natal
P04	100.000	Natal

Projeto 1

Cod_proj	Orçamento	Cidade
P01	100.000	Fortaleza
P04	100.000	Natal

$\sigma_{\text{orçamento} = 100.000}(\text{Projeto})$

Projeto 2

Cod_proj	Orçamento	Cidade
P03	80.000	Natal

$\sigma_{\text{orçamento} > 100.000}(\text{Projeto})$

Esta fragmentação não é completa!
Projeto1 UNION Projeto2 \neq Projeto



Fragmentação Vertical

- Um fragmento vertical conserva apenas alguns atributos da relação
- É necessário incluir a chave primária ou alguma chave candidata em todos os fragmentos verticais para que a relação completa possa ser reconstruída
- Um fragmento vertical pode ser especificado por: $\pi_{Li}(R)$



Fragmentação Vertical - Exemplo

Projeto 1: Informações sobre os orçamentos dos projetos
 Projeto 2: Informações sobre a localização dos projetos

} Fragmentos

Projeto

Cod_proj	Orçamento	Cidade
P01	100.000	Fortaleza
P02	120.000	Recife
P03	80.000	Natal
P04	100.000	Natal

Projeto 1

Cod_proj	Orçamento
P01	100.000
P02	120.000
P03	80.000
P04	100.000

Projeto 2

Cod_proj	Cidade
P01	Fortaleza
P02	Recife
P03	Natal
P04	Natal

$\pi_{\text{código, orçamento}}(\text{Projeto})$

$\pi_{\text{código, cidade}}(\text{Projeto})$



Fragmentação Vertical

- Um conj. de fragmentos verticais cujas listas de projeção L_1, L_2, \dots, L_n incluem todos os atributos de R e compartilham apenas a chave primária de R, é chamado de fragmentação vertical completa
- As listas de projeção satisfazem as seguintes condições:
 - $L_1 \cup L_2 \cup \dots \cup L_n = \text{Atributos}(R)$
 - $L_i \cap L_j = \text{PK}(R)$ para qualquer $i \neq j$, onde:
 - $\text{Atributos}(R)$ é o conjunto de atributos de R
 - $\text{PK}(R)$ é a chave primária de R



Fragmentação Vertical

- Para reconstruir R a partir de uma fragmentação vertical completa deve ser aplicada a operação de OUTER UNION se não houver fragmentação horizontal
- Para reconstruir R a partir de uma fragmentação vertical completa deve ser aplicada a operação de FULL OUTER JOIN se houver fragmentação horizontal



Fragmentação Vertical - Exemplo

Projeto

Cod_proj	Orçamento	Cod_coordenador	Cod_área	Cidade
P01	100.000	E01	A01	Fortaleza
P02	120.000	E03	A01	Recife
P03	80.000	E04	A03	Natal
P04	100.000	E05	A02	Natal

Projeto 1

Cod_proj	Orçamento
P01	100.000
P02	120.000
P03	80.000
P04	100.000

$\pi_{\text{cod_proj, orçamento}}(\text{Projeto})$

Projeto 2

Cod_proj	Cod_área
P01	A01
P02	A01
P03	A03
P04	A02

$\pi_{\text{cod_proj, cod_area}}(\text{Projeto})$

Esta fragmentação não é completa!
 Projeto1 OUTER UNION Projeto2 \neq Projeto



Fragmentação Mista - Exemplo

Projeto 1: Informações sobre os orçamentos dos projetos da cidade de Natal

Projeto 2: Informações sobre a localização dos projetos que tem um orçamento \geq 100.000

Projeto

Cod_proj	Orçamento	Cidade
P01	100.000	Fortaleza
P02	120.000	Recife
P03	80.000	Natal
P04	100.000	Natal

Projeto 1

Cod_proj	Orçamento
P03	80.000
P04	100.000

$\pi_{\text{código, orçamento}}(\sigma_{\text{cidade}=\text{'Natal'}}(\text{Projeto}))$

Projeto 2

Cod_proj	Cidade
P01	Fortaleza
P02	Recife
P04	Natal

$\pi_{\text{código, orçamento}}(\sigma_{\text{orçamento} \geq 100.000}(\text{Projeto}))$



Fragmentação de Dados

- Esquema de fragmentação
 - Conjunto de fragmentos que incluem todos os atributos e tuplas do banco de dados
 - O BD pode ser reconstruído a partir dos fragmentos aplicando-se operações de OUTER UNION (OUTER JOIN) e UNION
- Esquema de alocação
 - Descreve a alocação dos fragmentos nos sites do sistema de BDD
 - Ocorre replicação quando um fragmento está armazenado em mais de um site



Replicação de dados

- Replicação
 - Total
 - Aumenta disponibilidade
 - Melhora desempenho das consultas globais
 - Pode ser prejudicial para operações de atualização (é preciso atualizar cada cópia)



Replicação de dados

- Replicação
 - Parcial
 - Alguns fragmentos são replicados outros não
 - Bastante utilizado em aplicações móveis
- Sem replicação
 - Cada fragmento é armazenado em um único site
 - Todos os fragmentos devem ser disjuntos



Alocação de dados

- Cada fragmento ou cópia deve ser atribuído a um site particular
- A escolha dos sites e o grau de replicação dependem dos interesses em desempenho e disponibilidade
- Exemplo:
 - Alta disponibilidade + poucas atualizações + transações submetidas em qualquer site = BD completamente replicado
- Encontrar uma solução ótima para a alocação de dados distribuídos é um problema complexo!

