Processamento de Consultas

Banco de Dados: Teoria e Prática

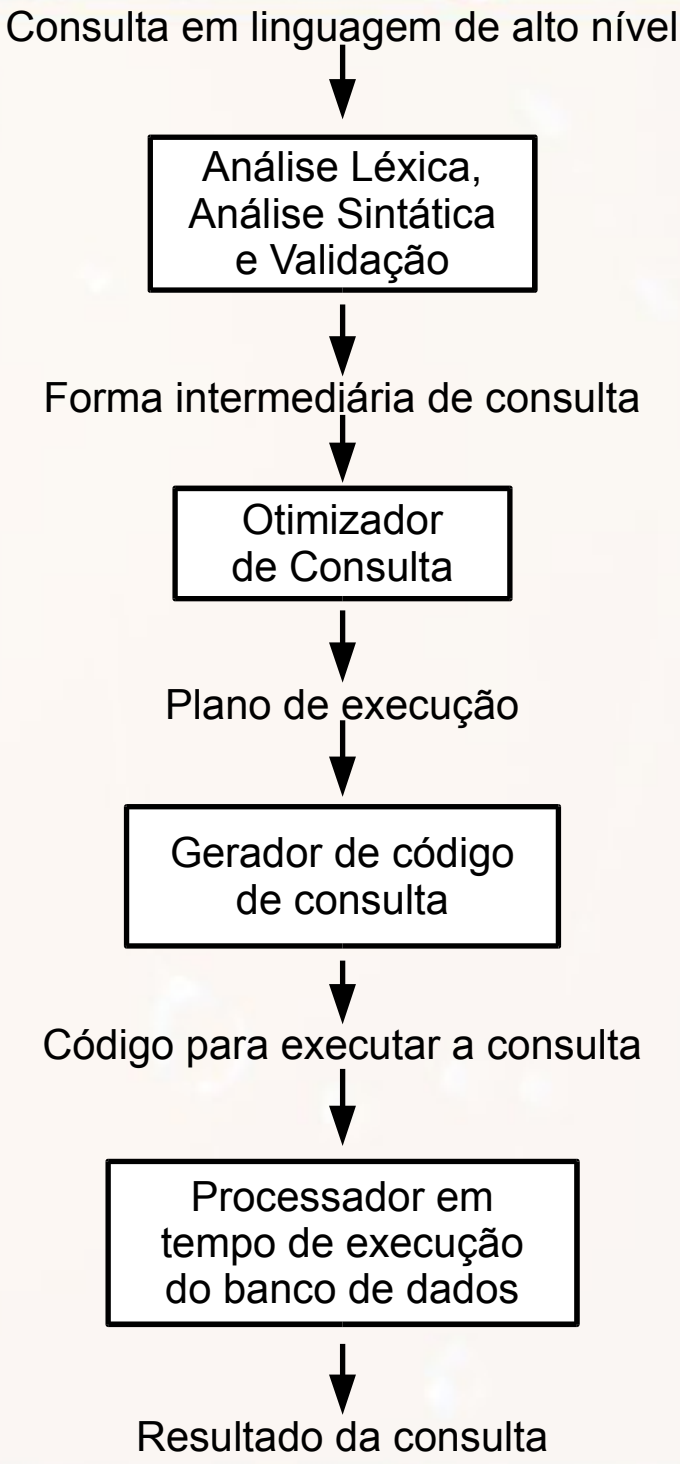
André Santanchè e Luiz Celso Gomes Jr
Instituto de Computação – UNICAMP
Setembro 2013

Execução de Consulta

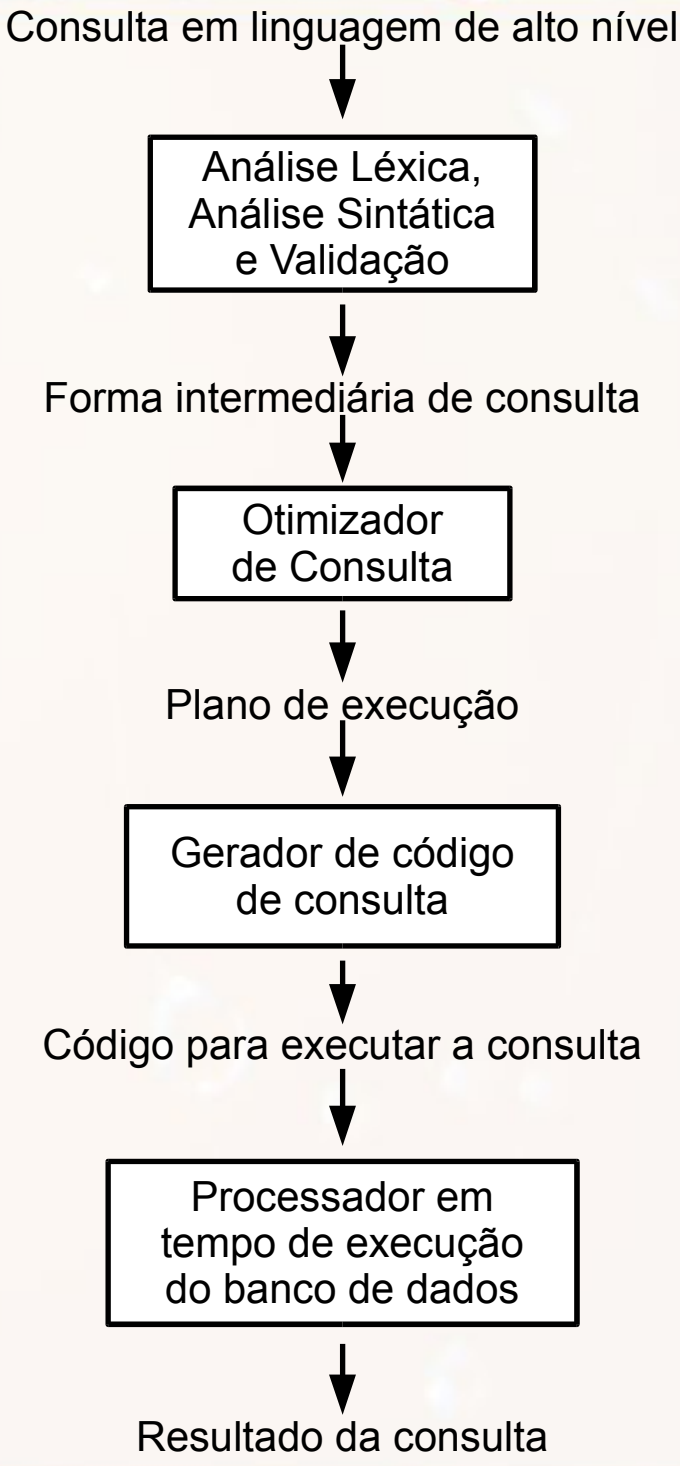
-

Passos Típicos

(Elmasri, 2010)

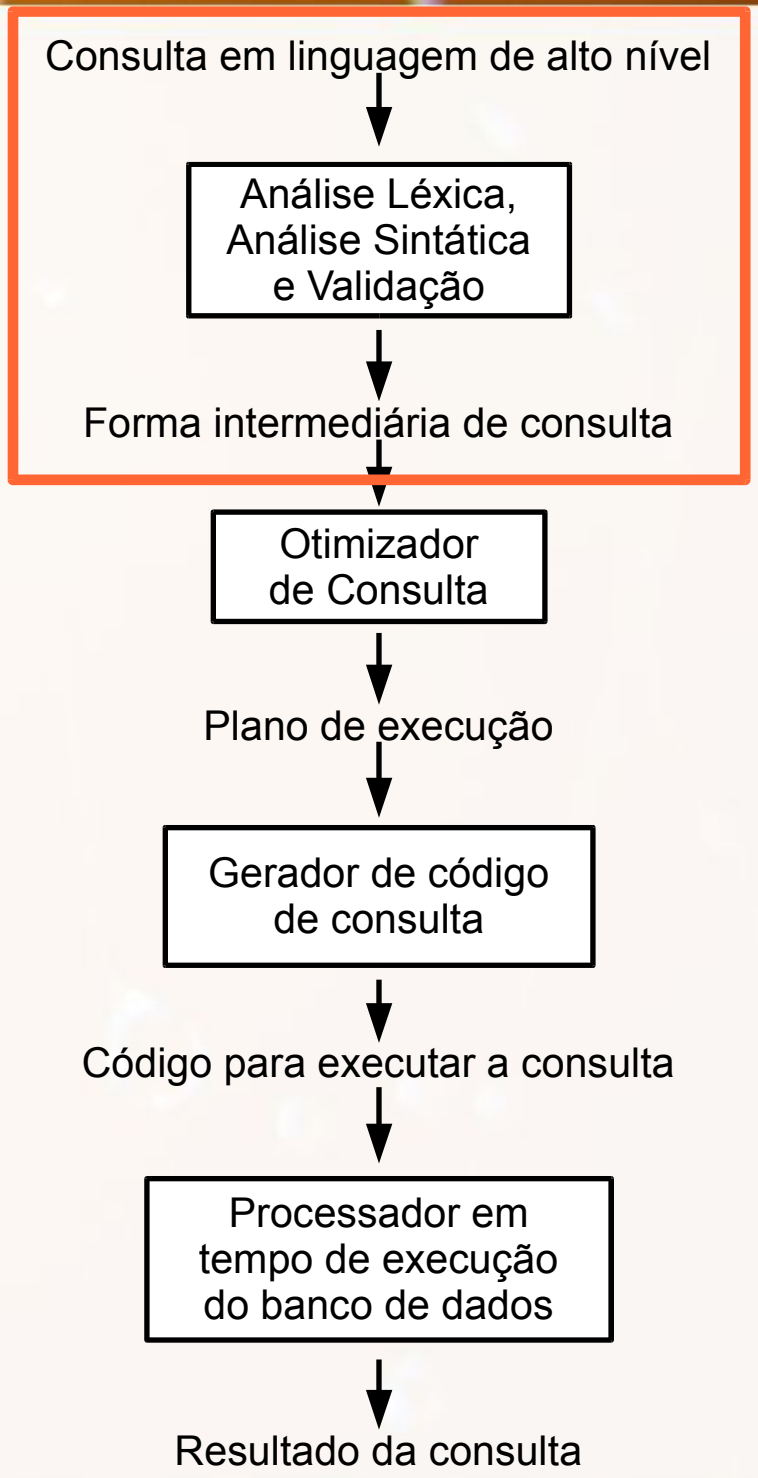


Execução de Consulta - Passos Típicos



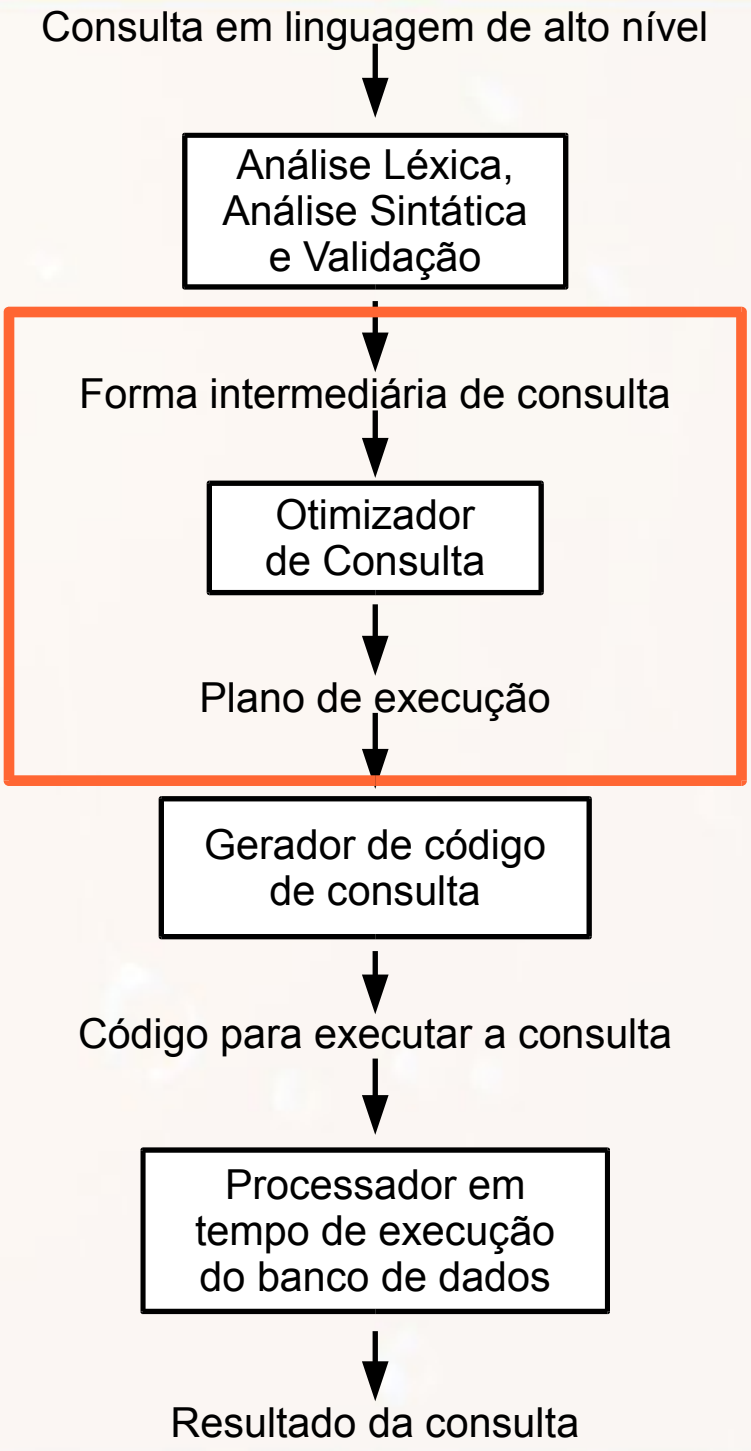
Análise e Validação

- Análise e Validação
  - Análise léxica
  - Análise sintática
  - Validação
- Representações internas:
  - árvore de consulta
  - grafo de consulta



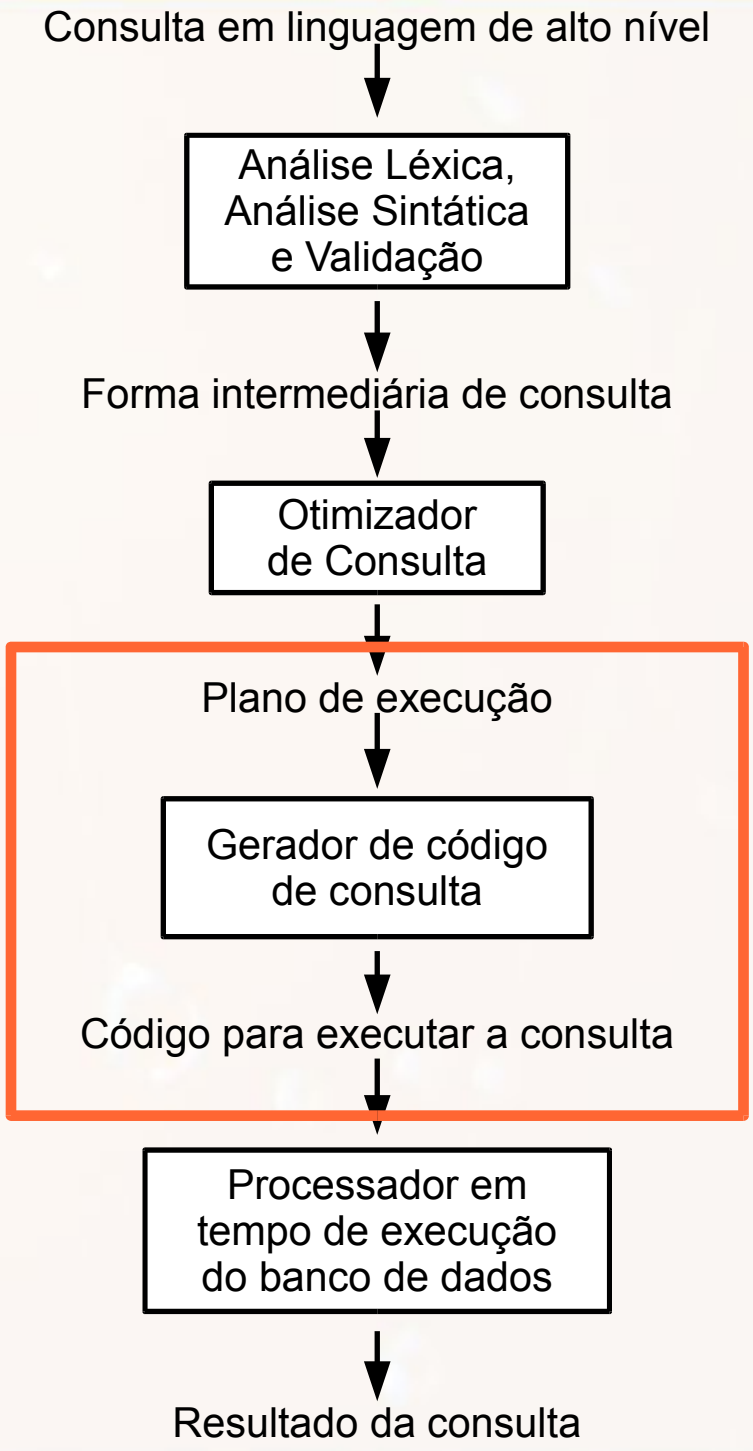
Estratégia de Execução

- Consulta possui muitas estratégias de execução possíveis
- Planejamento da Estratégia de Execução
  - Otimização → processo de escolha da estratégia adequada (razoavelmente eficiente)



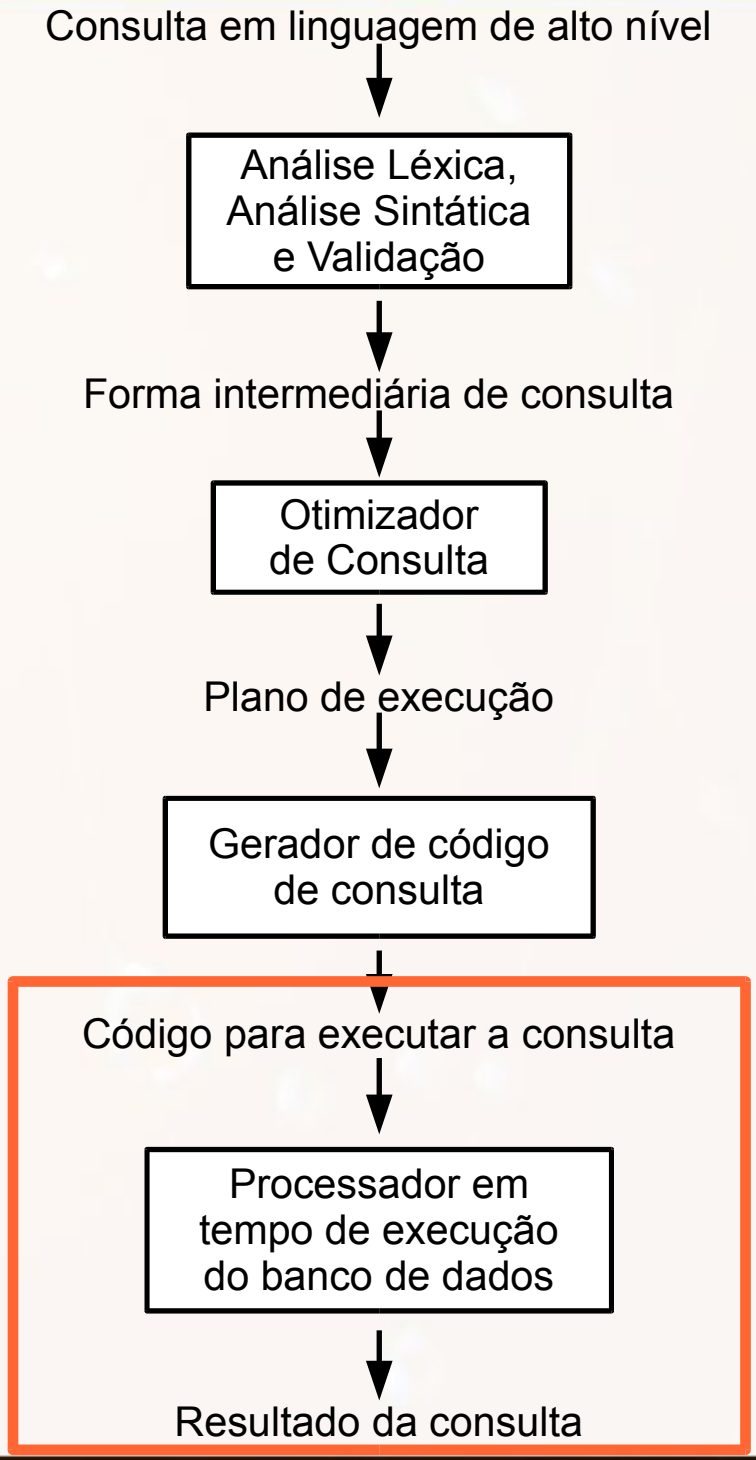
Código da Consulta

- Pode ser:
  - Executado diretamente
    - modo interpretado
  - Armazenado e executado quando necessário
    - modo compilado



Execução do Código

- Processador executa código da consulta
- Produz resultado da execução



Ênfase desta aula:
Otimização de Consultas

Consultas Declarativas

- “O quê” ao invés de “Como”
- Otimização de consulta
  - Solução razoavelmente eficiente (Elmasri, 2011)
  - Solução ótima pode ser muito custosa

Consulta SQL em Álgebra Relacional

- Consulta SQL → Álgebra Relacional Estendida
  - Inclui operadores como COUNT, SUM e MAX
- Consulta SQL decomposta em blocos
  - Bloco de Consulta ou Bloco Simples:
    - Contém uma única expressão SELECT-FROM-WHERE (GROUP BY e HAVING se houver)
    - Sem aninhamento
  - Consultas aninhadas são identificadas como consultas independentes

Decomposição em Blocos
Exemplo

- Tabela

```
Pessoa(Codigo, Nome, Telefone, AnoFiliacao)
```

- Nome dos filiados mais antigos:

```
SELECT Codigo, Nome
FROM PESSOA
WHERE AnoFiliacao = (SELECT MIN(AnoFiliacao))
                     FROM PESSOA)
```

- Blocos

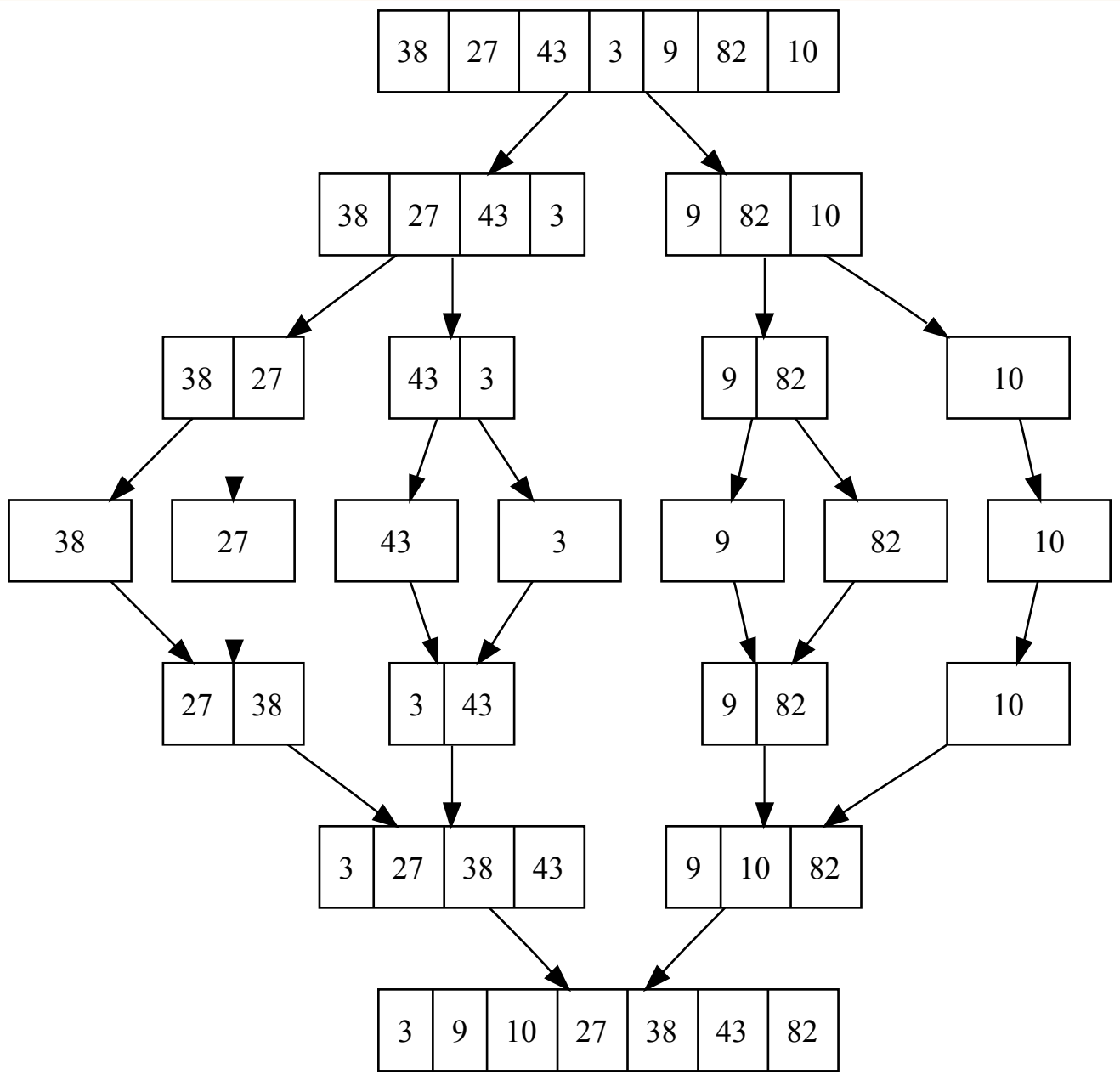
❶
```
SELECT Codigo, Nome
FROM PESSOA
WHERE AnoFiliacao = (referência ❷)
```

❷
```
SELECT MIN(AnoFiliacao))
FROM PESSOA
```

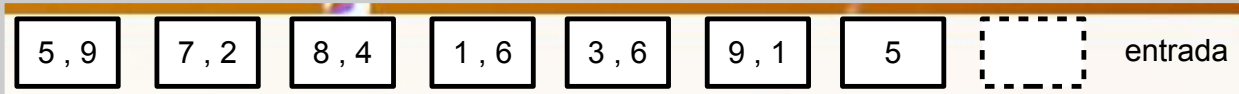
Algoritmos para Operações

Ordenação Externa

Merge Sort Tradicional



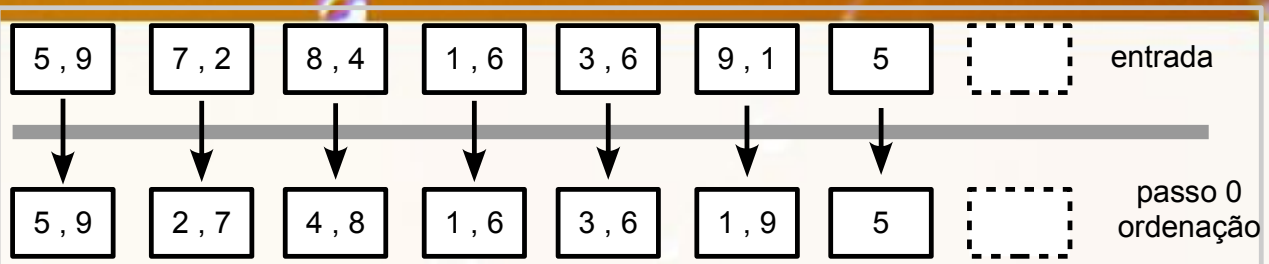
Ordenação Externa



Entrada organizada em páginas de tamanhos iguais:

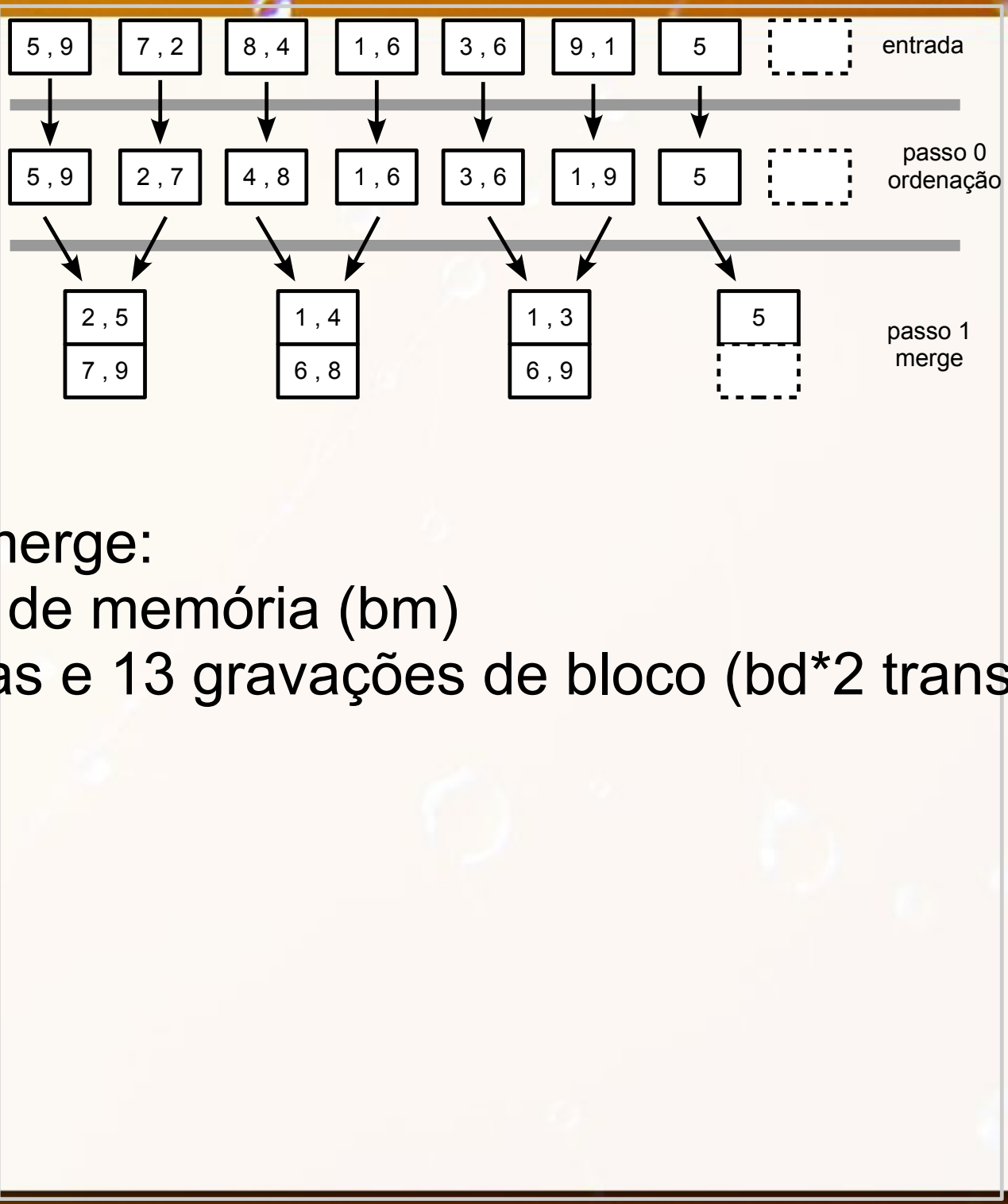
- 13 blocos de disco (bd)
- 3 blocos de memória (bm)

Exemplo Inspirado em (Ramakrishnan, 2013)



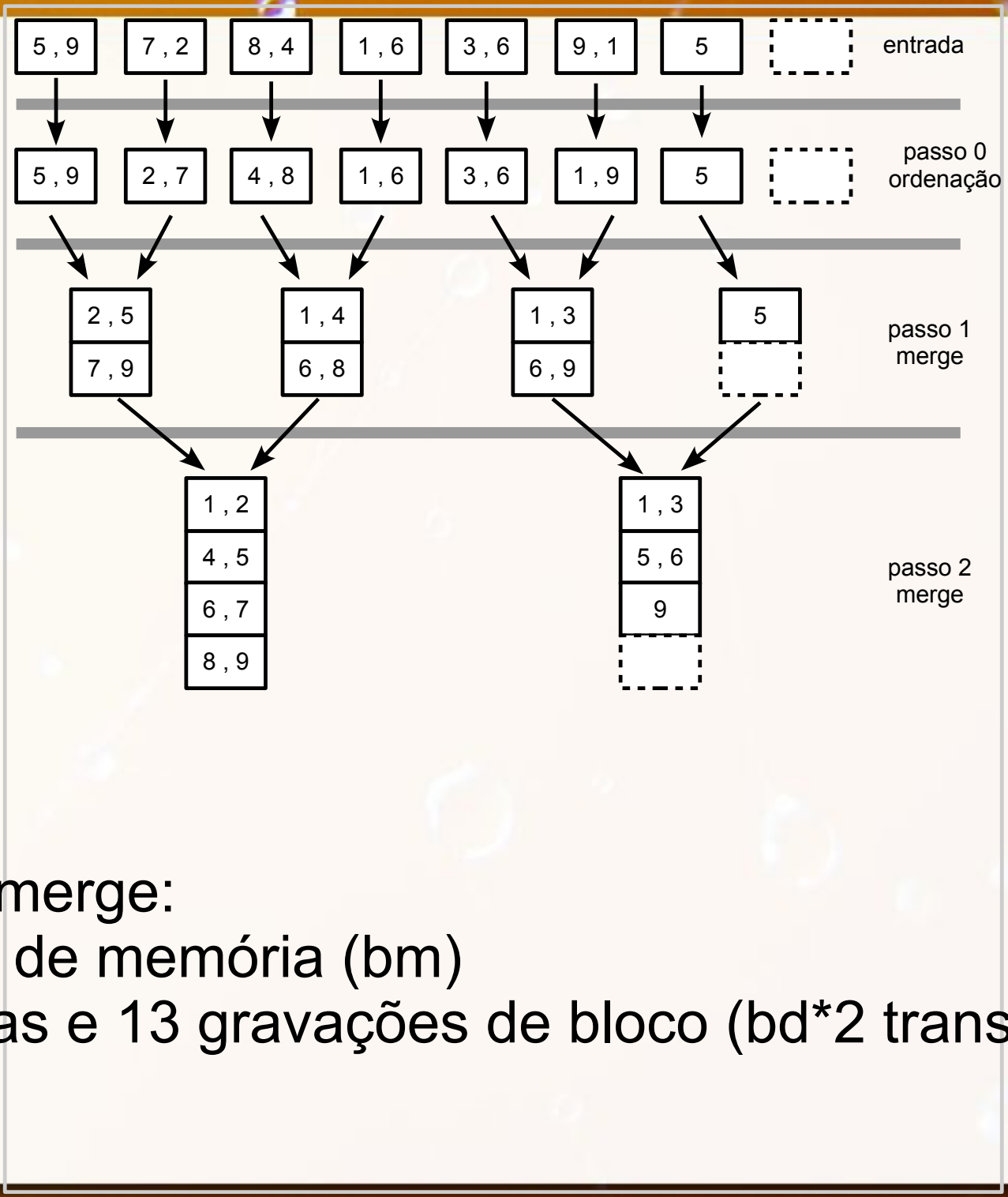
Passo inicial de ordenação de páginas em memória:

- pode ser usado qualquer algoritmo (e.g., quick sort)
- 13 leituras e 13 gravações de bloco (bd*2 transferências)



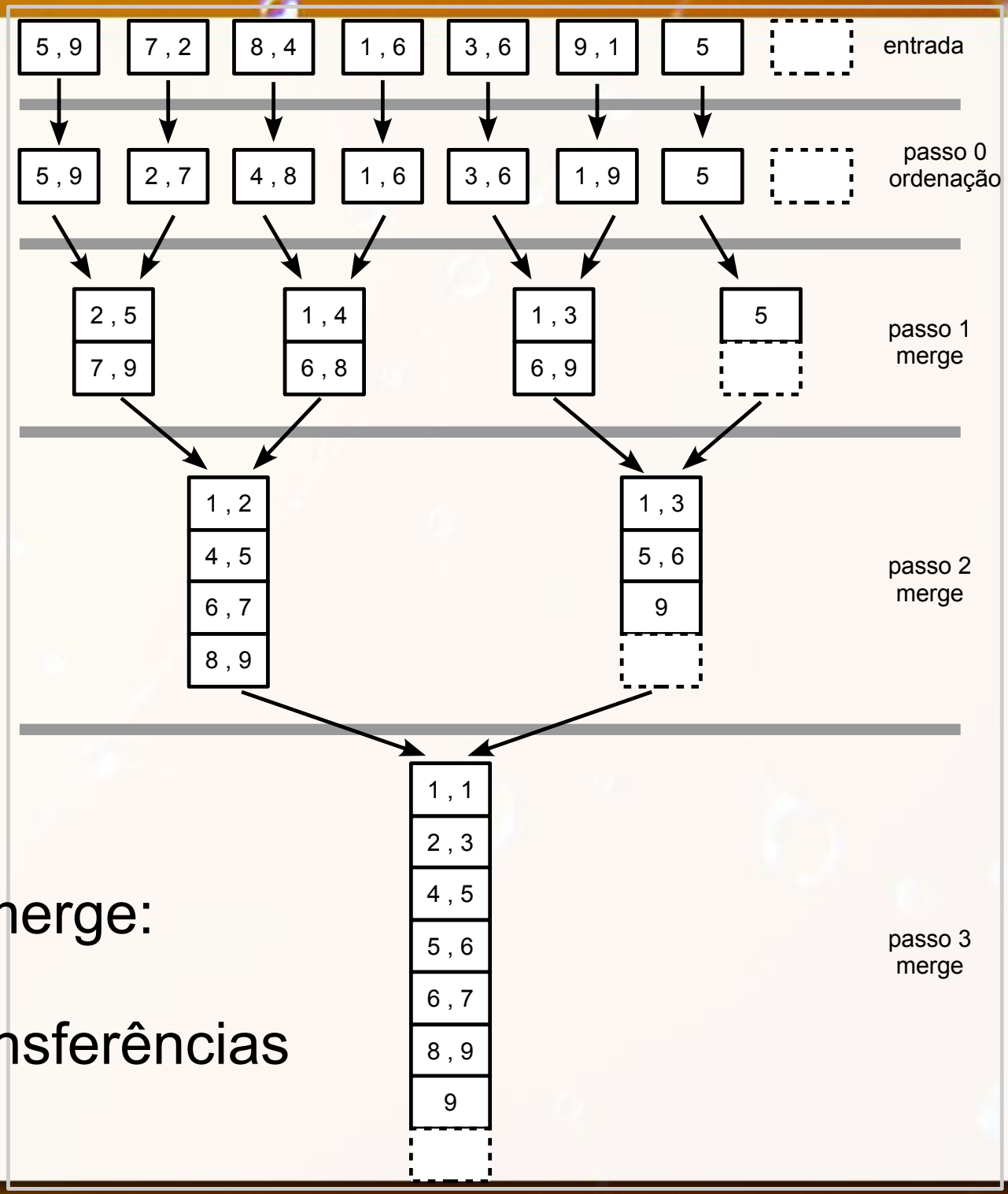
Primeiro merge:

- 3 blocos de memória (bm)
- 13 leituras e 13 gravações de bloco (bd*2 transferências)



Segundo merge:

- 3 blocos de memória (bm)
- 13 leituras e 13 gravações de bloco (bd*2 transferências)



Terceiro merge:

- 3 bms
- bd*2 transferências

Ordenação Externa
Números

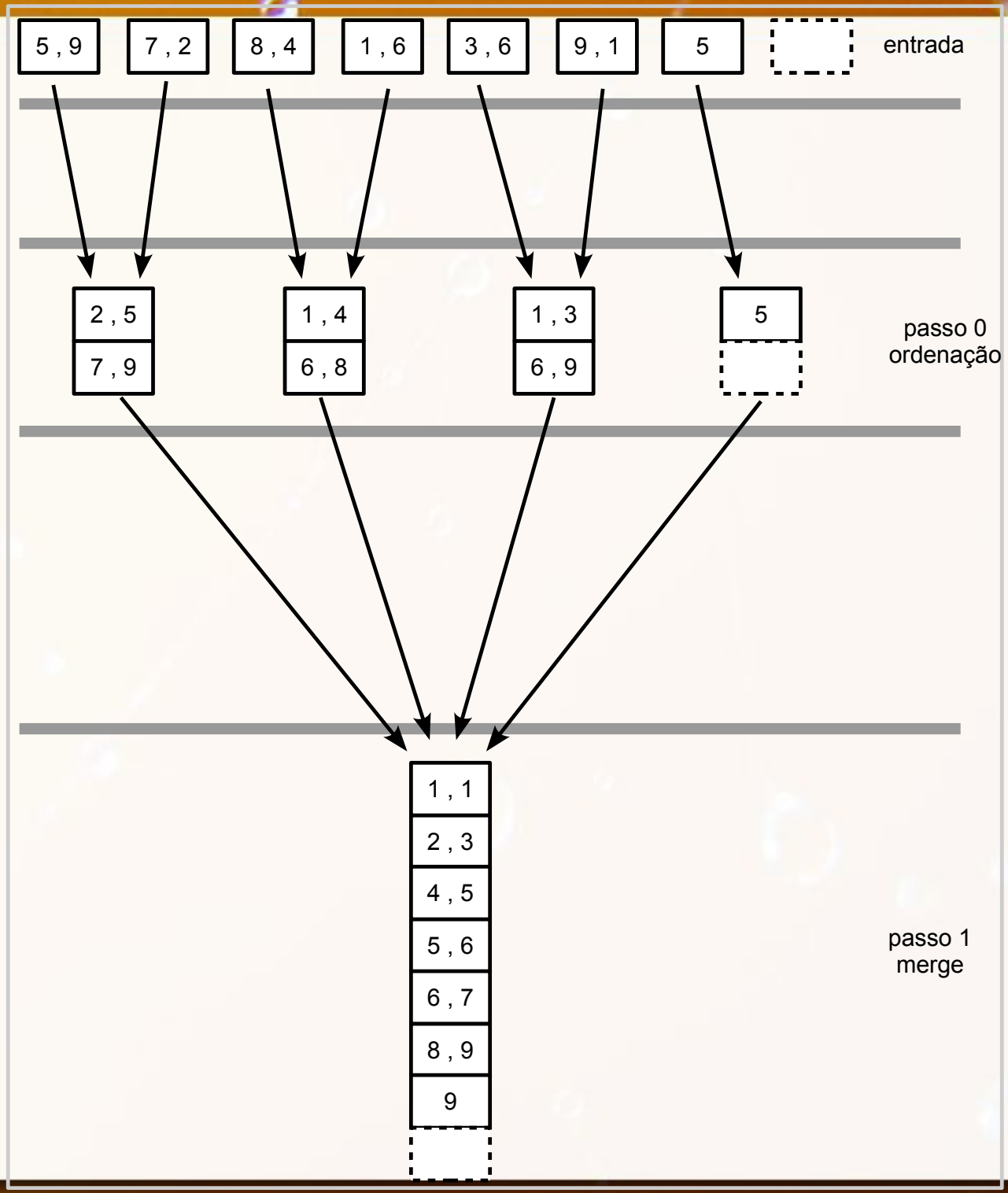
- bd – blocos em disco
- bm – blocos de memória
  - bm_e – blocos de entrada = bm - 1
  - bm_s – blocos de saída = 1

Ordenação Externa
Números

- Ordenação – passo 0
  - 2*bd = 2*13 = 26 transferências (leitura e gravação)
- Merge
  - 2*bd = 2*13 = 26 transferências a cada estágio
- Rodadas por nível
  - rodadas = $\lceil bd/bm_e \rceil = \lceil 13/2 \rceil = 8$
- Níveis
  - $\lceil \log_2 \text{rodadas} \rceil + 1 = \lceil \log_2 8 \rceil + 1 = 4$ níveis
- Custo: $2*bd * (\lceil \log_2 \text{rodadas} \rceil + 1)$

Como Otimizar?

Se eu tiver 5 blocos de memória?


5 , 9
7 , 2
8 , 4
1 , 6
3 , 6
9 , 1
5
entrada
2 , 5
7 , 9
1 , 4
6 , 8
1 , 3
6 , 9
5
passo 0
ordenação
1 , 1
2 , 3
4 , 5
5 , 6
6 , 7
8 , 9
9
passo 1
merge

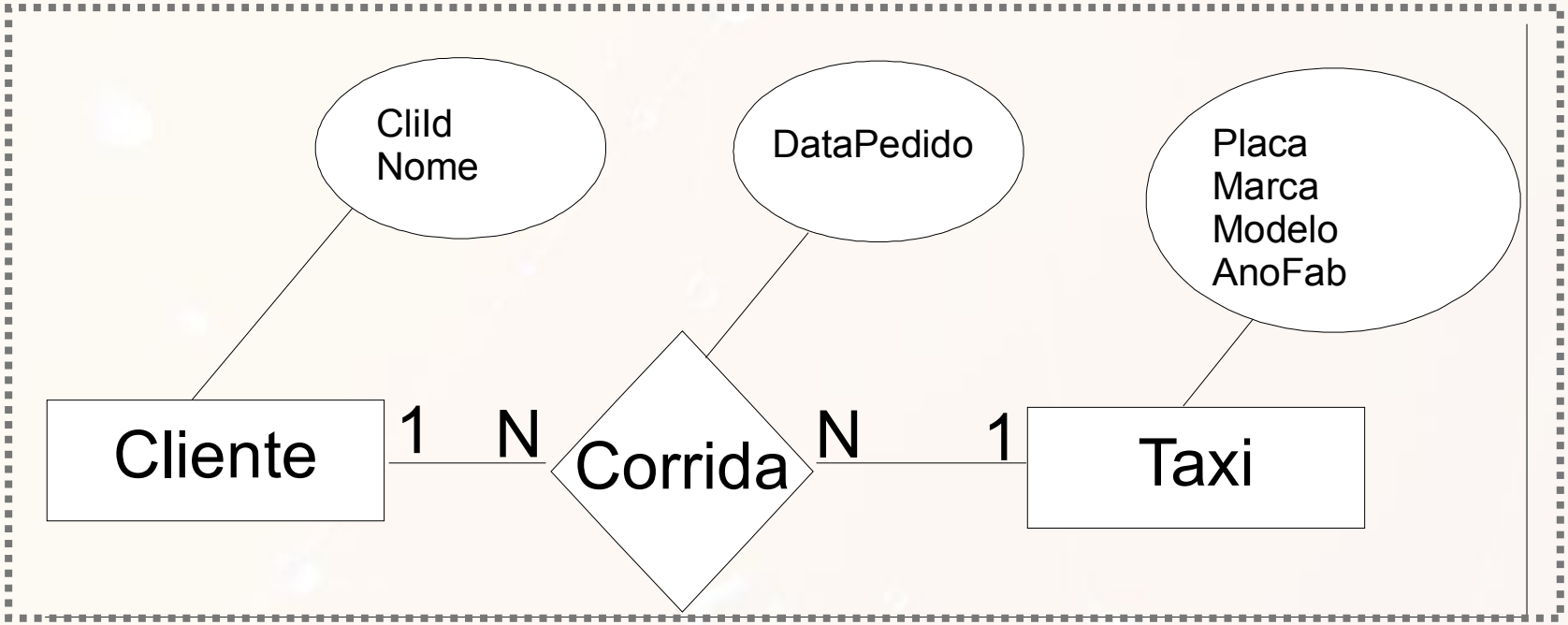
Ordenação Externa Números

- Ordenação – passo 0
  - 2*bd = 2*13 = 26 transferências (leitura e gravação)
- Merge
  - 2*bd = 2*13 = 26 transferências a cada estágio
- Rodadas por nível
  - rodadas = $\lceil bd/bm_e \rceil = \lceil 13/4 \rceil = 4$
- Rodadas (níveis)
  - $\lceil \log_{bme} rodadas \rceil + 1 = \lceil \log_4 4 \rceil + 1 = 2$ rodadas
- Custo: 2*bd * ($\lceil \log_{bme} rodadas \rceil + 1$)

Seleção

Esquema Conceitual – Exemplo

Táxis



Este é um subconjunto do Estudo de Caso proposto “Despacho e controle de Táxis via terminais móveis ligados on-line com um sistema multi-usuário” por prof. Geovane Cayres Magalhães

Tabelas para exemplo - Táxis

Táxi (TX)

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Corrida (R1)

ClId	Placa	DataPedido
1755	DAE6534	15/02/2003
1982	JDM8776	18/02/2003

Seleção?

$$\sigma_{Placa='JDM8776'}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Exatamente Igual
Chave Primária

$$\sigma_{Placa='JDM8776'}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Exatamente Igual
Outra Chave

$$\sigma_{AnoFab=2002}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Seleção?

$$\sigma_{AnoFab=2002}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Seleção?

$$\sigma_{AnoFab>2000}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Faixa (>, <, >=, <=)

$$\sigma_{AnoFab>2000}(TX)$$

Placa	Marca	Modelo	AnoFab
DAE6534	Ford	Fiesta	1999
DKL4598	Wolksvagen	Gol	2001
DKL7878	Ford	Fiesta	2001
JDM8776	Wolksvagen	Santana	2002
JJM3692	Chevrolet	Corsa	1999

Algoritmos de Seleção

- Exatamente igual
  - chave primária
  - outra chave
- >, <, >=, <=
- compostos

Algoritmos de Seleção

- Pesquisa linear
- Pesquisa binária
- Usando índice primário
- Usando chave hash
- Combinado com o índice primário
- Usando índice de agrupamento
- Usando índice secundário

Seleção Conjuntiva x Dijuntiva

- seleção conjuntiva - e.g., and
- seleção dijuntiva - e.g., or

Algoritmos de Seleção Conjuntiva

- Índice para uma das condições
- Índice composto envolvendo ambas as condições
- Índice individual para cada condição

Seletividade

- seletividade: valor entre 0 e 1
- n registros
- igualdade atributo único
  - seletividade: 1/n

Seletividade
Atributo Não Único

- i valores
- i igualmente distribuído
- registros por valor?
- seletividade?

Seletividade
Atributo Não Único

- i valores
- i igualmente distribuído
- n/i registros por valor
- seletividade: 1/i

Seletividade
Atributo Não Único

- primeiro as condições com valor menor de seletividade

Exercício 1

- Considere a execução de uma consulta envolvendo uma seleção em um atributo que possui um índice. É sempre mais eficiente usar o índice do atributo no processamento?

Junção
(Join)

Junção (Join) de Loop Aninhado

```
for each ti
   for each tj
      if match(ti, tj)
         add-result(ti, tj)
```

Junção de Loop Aninhado Números

- ni - número de tuplas ti
- nj - número de tuplas tj
- pares de tuplas? (comparações?)

Junção de Loop Aninhado Números

- ni - número de tuplas ti
- nj - número de tuplas tj
- ni*nj - pares de tuplas

Junção de Loop Aninhado Números

- ni - número de tuplas ti
- nj - número de tuplas tj
- ni*nj - pares de tuplas

- bi - bloco de tuplas ti
- bj - bloco de tuplas tj
- leituras de blocos?

Junção de Loop Aninhado Números

- ni - número de tuplas ti
- nj - número de tuplas tj
- ni*nj - pares de tuplas

- bi - bloco de tuplas ti
- bj - bloco de tuplas tj
- bi + bj*ni leituras de blocos

Junção de Loop Aninhado Números

- Situações:
  - Quantas transferências de bloco se todos os blocos estiverem na memória?
  - Quantas transferências se os blocos de um dos loops estiver todo na memória e qual deles escolher (bi ou bj)

Junção de Loop Aninhado Números

- Situações:
  - Quantas transferências de bloco se todos os blocos estiverem na memória?
    - bi + bj transferências
  - Quantas transferências se os blocos de um dos loops estiver todo na memória e qual deles escolher (bi ou bj)?
    - escolher bj
    - bi + bj transferências

Junção de Loop Aninhado em Bloco

```
for each bi
  for each bj
    for each ti in bi
      for each tj in bj
        if match(ti, tj)
          add-result(ti, tj)
```

Junção de Loop Aninhado em Bloco Números

- bi - bloco de tuplas ti
- bj - bloco de tuplas tj
- leituras de blocos?

Junção de Loop Aninhado em Bloco Números

- bi - bloco de tuplas ti
- bj - bloco de tuplas tj
- bi + bj*bi leituras de blocos

Exercício 2

- Considere as seguintes tabelas e consulta:
  - `Aluno(ra, nome, id_dept)`
  - `Departamento(id_dept, nome_dept)`
  - ```
    SELECT ra, nome, nome_dept
    FROM Aluno, Departamento
    WHERE Aluno.id_dept = Departamento.id_dept
    ```
- Escreva o pseudo-código para o processamento do join na consulta acima.

a) Considere que todas as tabelas cabem na memória.

b) Considere que apenas a tabela Departamento cabe na memória.
    ```

Outras Junções

- Junção Indexada
- Junção Merge
- Junção Hash

Projeção

- Recorte dos campos
- (?)

Projeção

- Recorte dos campos
- Registros sem duplicatas
  - SQL → padrão não eliminar duplicatas
    - DISTINCT → elimina duplicatas
  - Registros com garantia de ser únicos
    - e.g., contendo chave primária
  - Registros sem garantia de ser únicos
    - ordenação
    - hashing

Otimização de Consulta

SQL p/ Álgebra

- Versão SQL

```
SELECT Codigo, Nome
FROM PESSOA
WHERE AnoFiliacao = 1990
```

- Versão em álgebra

$$\pi_{Codigo,Nome}(\sigma_{AnoFiliacao=1990}(PESSOA))$$

- Versão Árvore

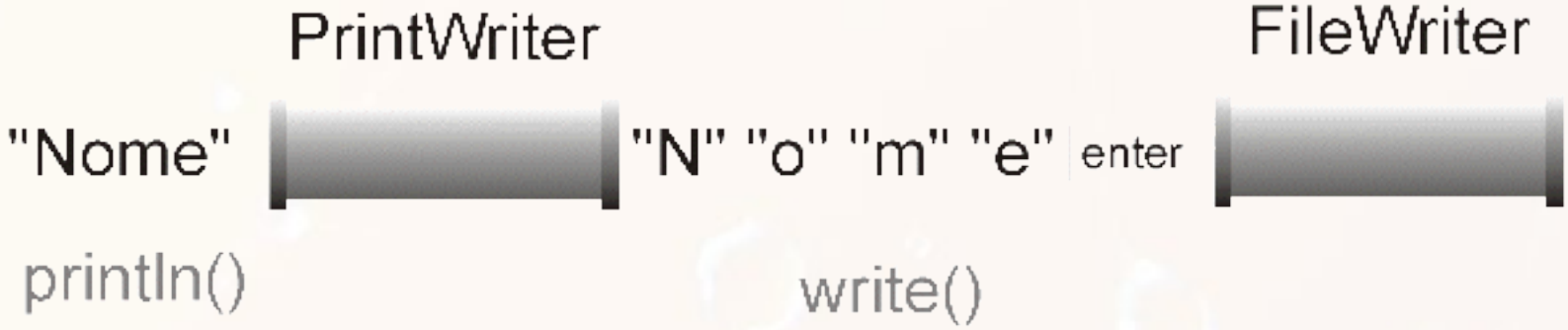
Combinação de Operações usando Pipelining

- Uma consulta é mapeada em uma sequência de operações
- A execução de cada operação produz um resultado temporário
- Alternativa
  - Evitar ao máximo resultados temporários
  - Pipelining
    - concatena operações
    - conforme uma saída é produzida gera entrada para a operação subsequente

Pipelining

Pattern Pipe & Filter

exemplo: Java Writer



Exemplo de Pipeline

PESSOA

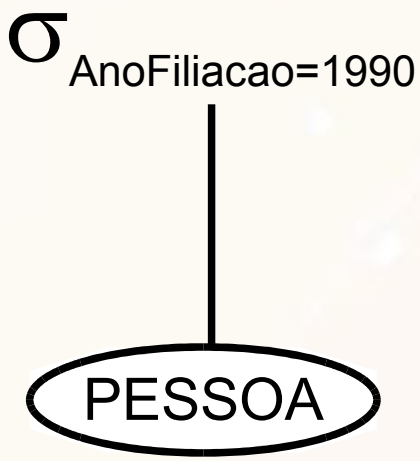
Codigo	Nome	Telefone	AnoFiliacao
1525	Asdrúbal	5432-1098	1990
1637	Doriana	9876-5432	1983
1701	Quincas	8765-4321	1985
2042	Melissa	7654-3210	1990
2111	Horácio	6543-2109	1983

Exemplo de Pipeline

PESSOA

Codigo	Nome	Telefone	AnoFiliacao
1525	**Asdrúbal**	**5432-1098**	**1990**
1637	Doriana	9876-5432	1983
1701	Quincas	8765-4321	1985
2042	**Melissa**	**7654-3210**	**1990**
2111	Horácio	6543-2109	1983

Exemplo de Pipeline



PESSOA

Codigo	Nome	Telefone	AnoFiliacao
1525	Asdrúbal	5432-1098	1990
2042	Melissa	7654-3210	1990

Exemplo de Pipeline

PESSOA

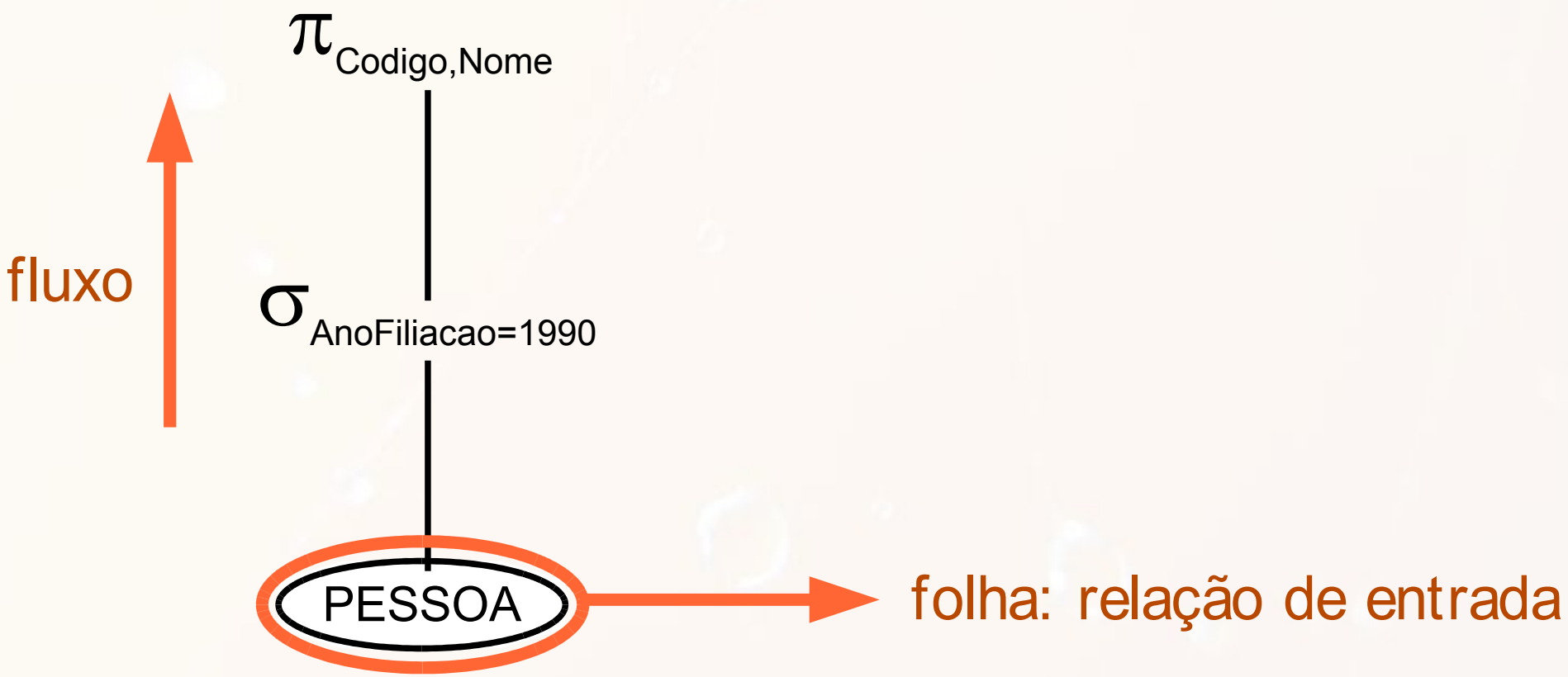
Codigo	Nome	Telefone	AnoFiliacao
1525	Asdrúbal	5432-1098	1990
2042	Melissa	7654-3210	1990

Exemplo de Pipeline

PESSOA

Codigo	**Nome**
1525	Asdrúbal
2042	Melissa

Árvore de Consulta



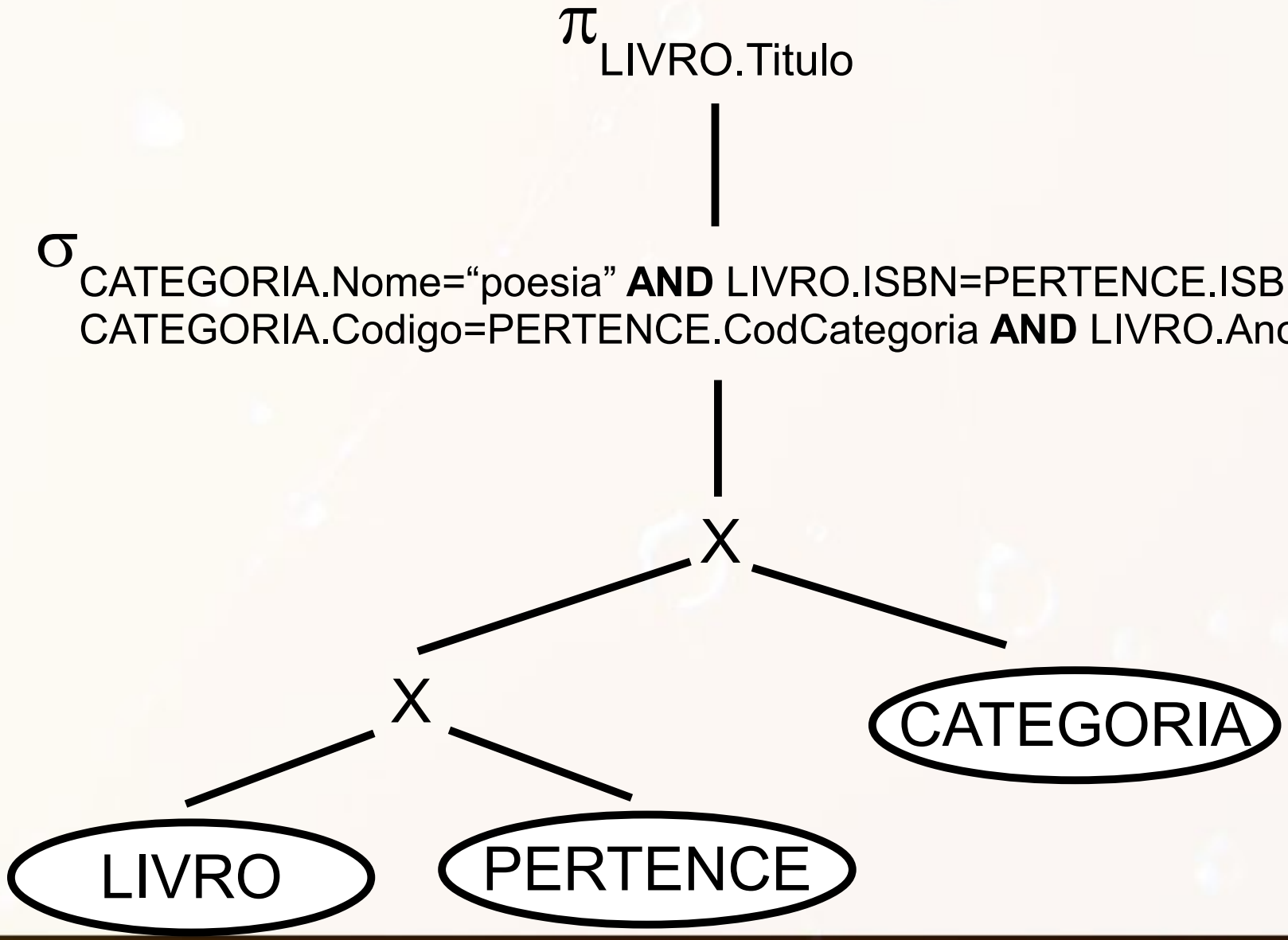
Heurísticas para Otimização de Consulta (Elmasri, 2011)

Heurísticas para Otimização de Consulta

- Título dos livros sobre poesia escritos depois de 1996

```
SELECT LIVRO.Titulo
FROM   LIVRO, PERTENCE, CATEGORIA
WHERE  CATEGORIA.Nome = "poesia" AND
       LIVRO.ISBN = PERTENCE.ISBN AND
       CATEGORIA.Codigo = PERTENCE.CodCategoria AND
       LIVRO.Ano > 1996
```

Heurística para Otimização de Consulta



Regras de Transformação

1. Operações seleção conjuntivas podem se converter em cascatas de seleção
2. Operação de seleção é comutativa
3. Comutação de seleção com projeção
   - caso o resultado da projeção tenha atributos requeridos pela seleção

Regras de Transformação

4. Seleção e junção (ou produto cartesiano) são comutativas
   - se atributos da seleção são de apenas uma das relações
5. Operações de união e interseção são comutativas
   - diferença não é

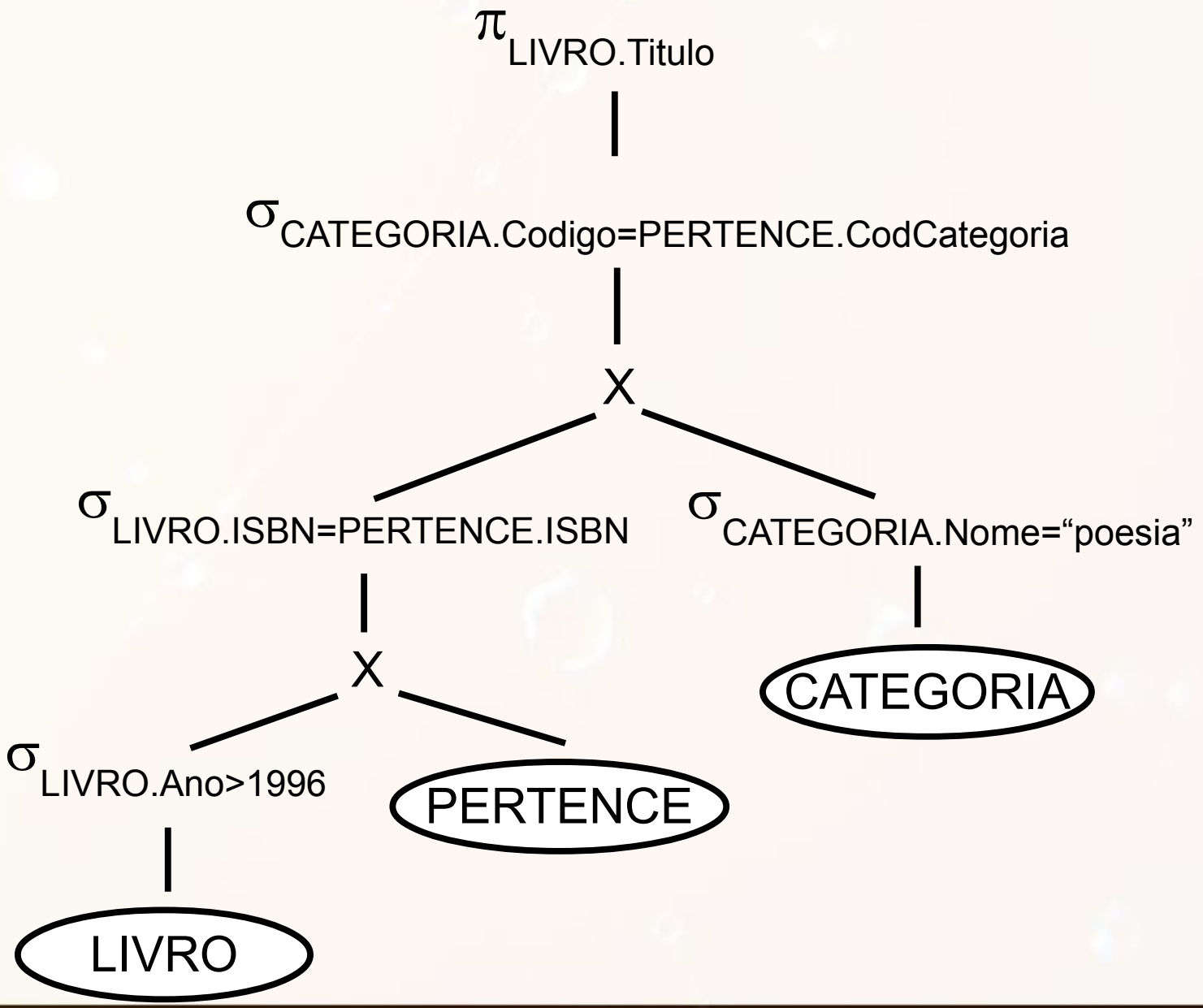
Regras de Transformação

6. Seleção é comutativa com operações de conjunto (união, interseção e diferença)
   - sel (A @ B) equivale sel(A) @ sel(B)

Heurísticas

- Quebrar operações de seleção conjuntivas (1)
  - maior liberdade
- Mover seleção em direção às folhas (2), (3), (4), (5) e (6)
  - apenas 1 tabela → acima da tabela
  - duas tabelas → acima da junção

Quebrando e Descendo Seleções



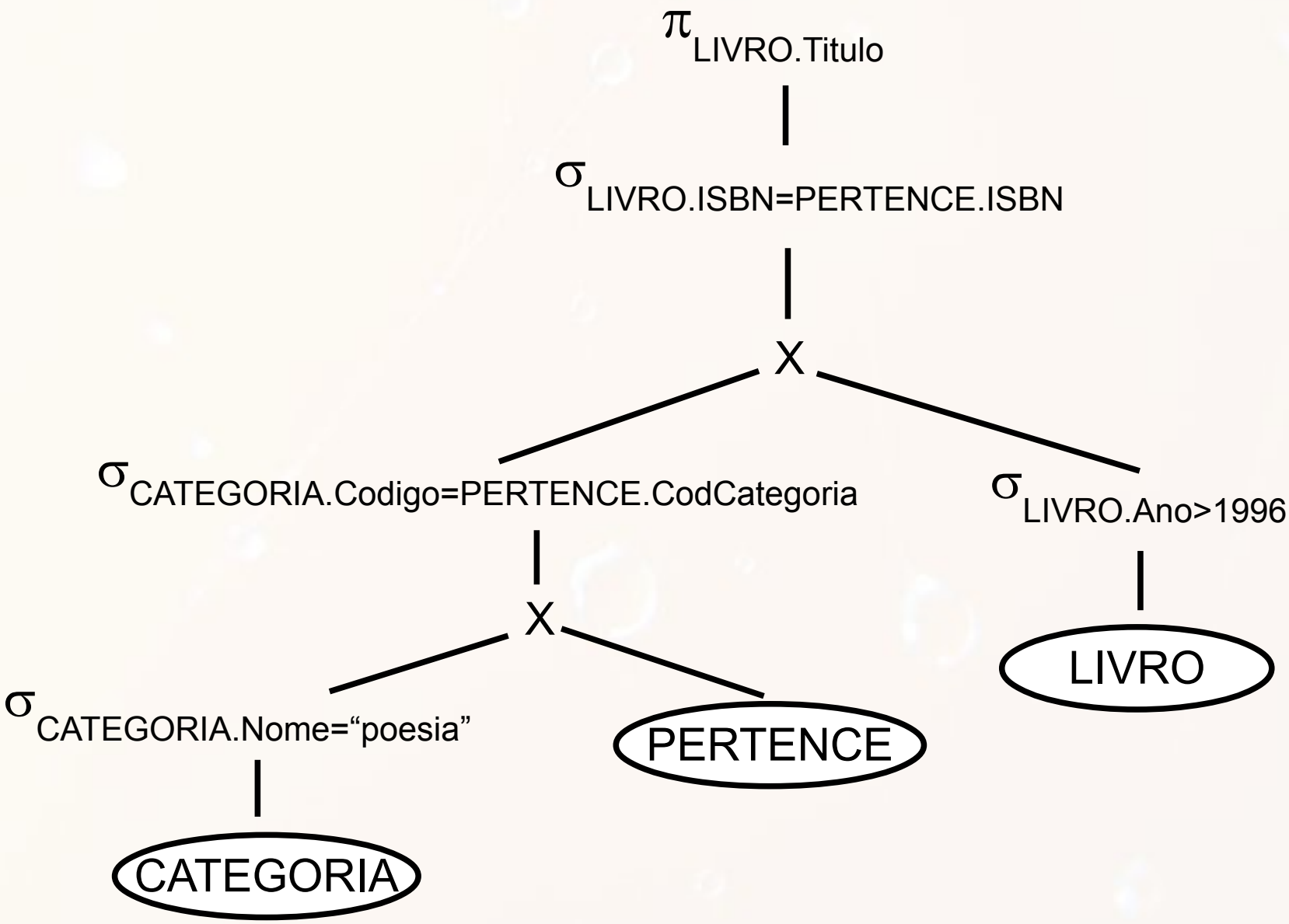
Regras de Transformação

7. As operações de junção e produto cartesiano são comutativas
8. As operações de junção, produto cartesiano, união e interseção são associativas

Heurística

- Operações de seleção mais restritivas devem ser executadas primeiro (5) e (6)

Troca de Categoria com Livro



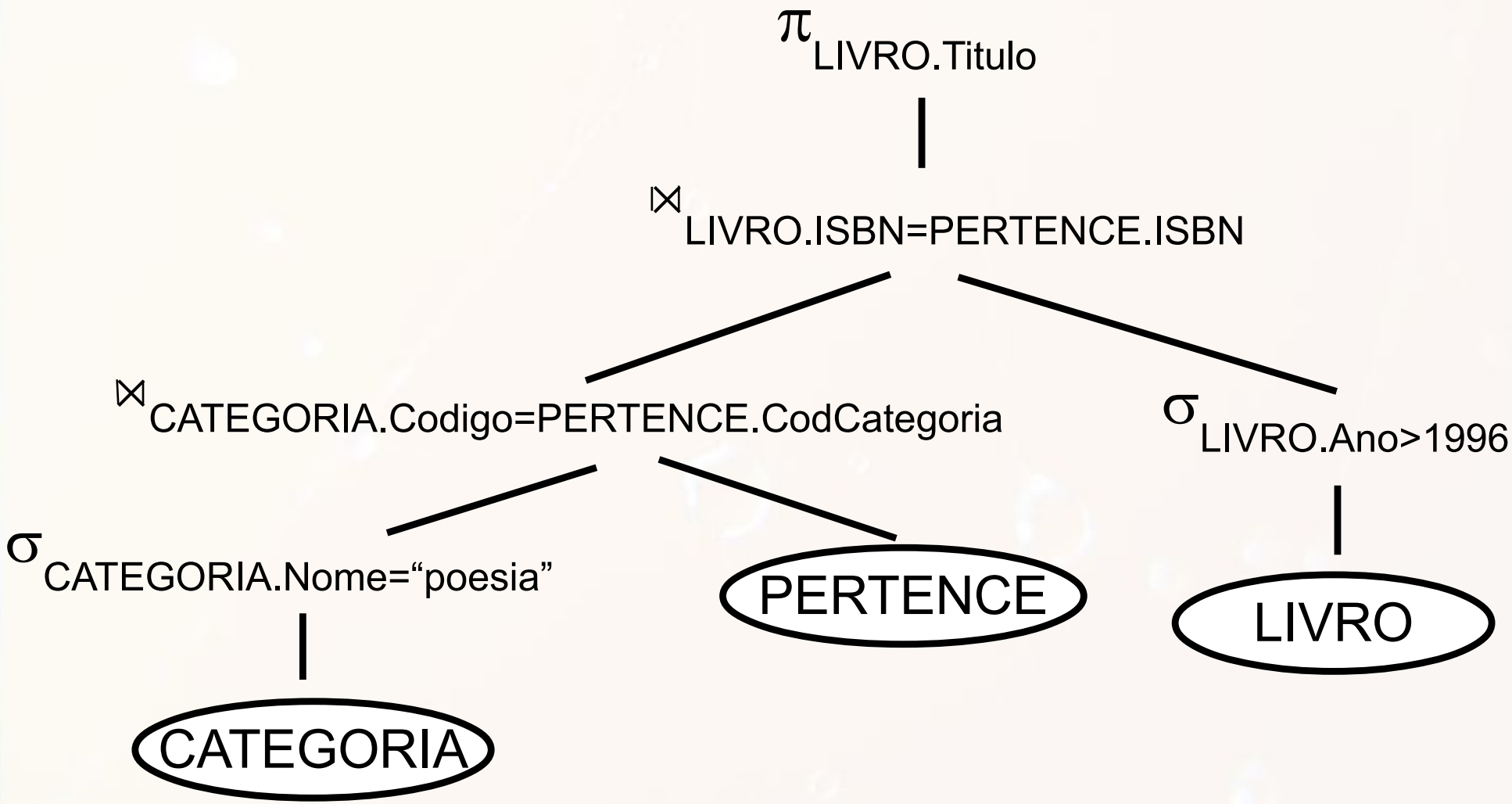
Regra de Transformação

9. Operações de produto cartesiano + seleção podem se converter em junção

Heurística

- Converta produtos cartesianos + seleções em junções

Produto Cartesiano + Seleção = Junção



Regras de Transformação

10. Cascata de projeções podem ser ignoradas e convertidas na última
    - Pr1(Pr2(Pr3(A))) equivale Pr1(A)
11. Operações de projeção e união são comutativas
    - proj (A U B) equivale proj(A) U proj(B)

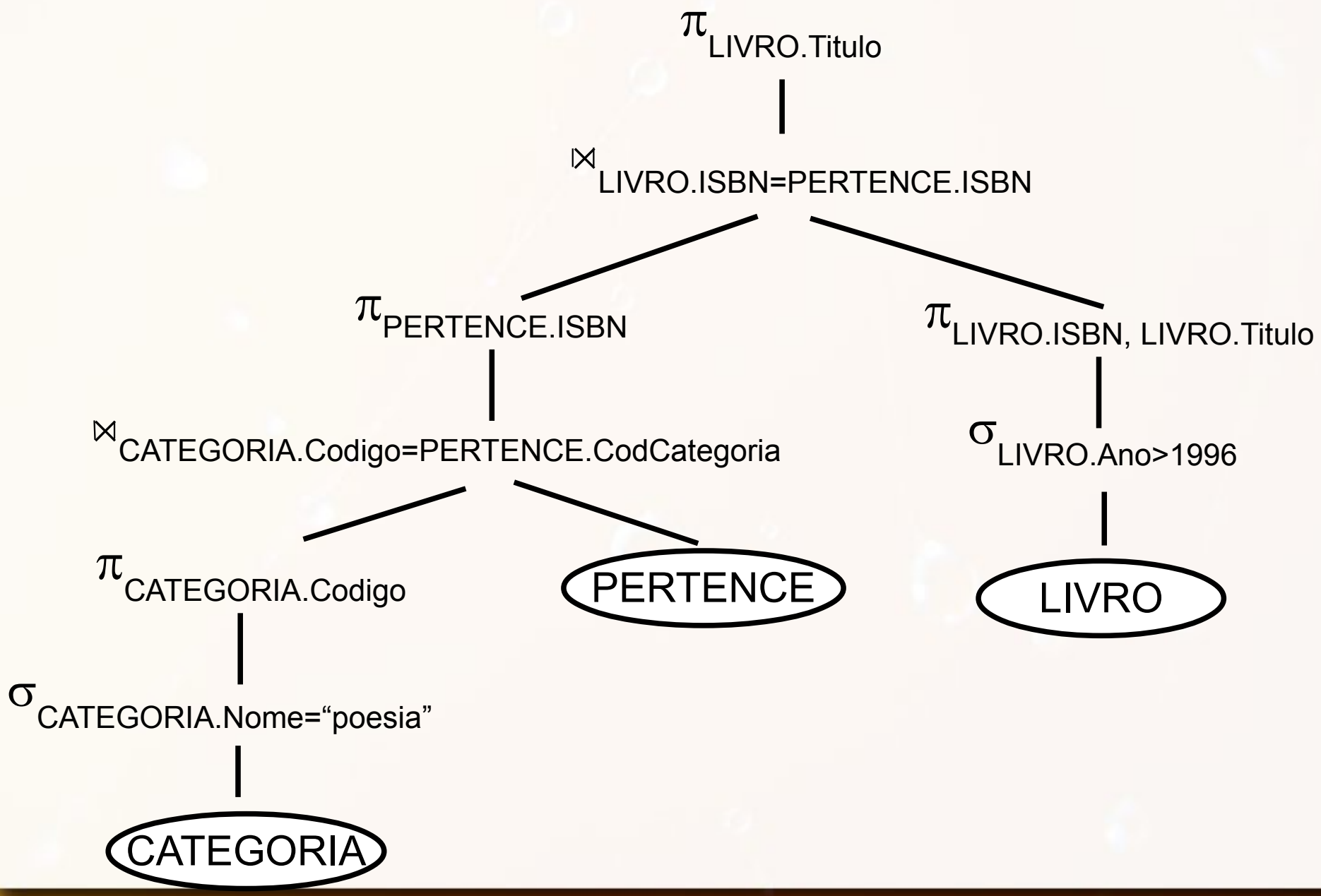
Regras de Transformação

12. Operação de projeção pode ser comutada com junção (ou produto cartesiano)
    - Relação A → atributos $a_1,\ldots,a_n$
    - Relação B → atributos $b_1,\ldots,b_m$
    - L = $(a_1,\ldots,a_n,b_1,\ldots,b_m)$
    - Condição só contém atributos L
    - $proj_L$(A junção B) equivale ($proj_{a1,\ldots,an}$(A)) junção ($proj_{b1,\ldots,bn}$(B))

Heurística

- Baseados em (10), (11) e (12)
  - Desmembrar operações de projeção
  - Mover projeções em direção às folhas
  - Criar operações de projeção para manter apenas atributos necessários

Projeções Mais Cedo



Heurística

- Identificar subárvores com operações a ser combinadas em um algoritmo

Exercício 3

- Considere as seguintes tabelas:
  - R(A,B,C,D)
  - S(E,F,G,H) - E é chave-estrangeira que referencia R(A)

a) desenhe um plano de acesso otimizado para a consulta:

  - ```
    select A from R, S
    where A=5 and G=7 and F=A
    ```
    ```

Referências

- Elmasri, Ramez; Navathe, Shamkant B. (2005) **Sistemas de Bancos de Dados**. Addison-Wesley, 4ª edição em português.
- Elmasri, Ramez; Navathe, Shamkant B. (2011) **Sistemas de Bancos de Dados**. Addison-Wesley, 6ª edição em português.
- Ramakrishnan, Raghu; Gehrke, Johannes (2003) **Database Management Systems.** McGraw-Hill, 3rd edition.

André Santanchè
http://www.ic.unicamp.br/~santanche

Licença

- Estes slides são concedidos sob uma Licença Creative Commons. Sob as seguintes condições: Atribuição, Uso Não-Comercial e Compartilhamento pela mesma Licença.
- Mais detalhes sobre a referida licença Creative Commons veja no link:
  http://creativecommons.org/licenses/by-nc-sa/3.0/

- Fotografia da capa e fundo por
  http://www.flickr.com/photos/fdecomite/
  Ver licença específica em
  http://www.flickr.com/photos/fdecomite/1457493536/