

FUNDAMENTOS DE BANCO DE DADOS BANCO DE DADOS



Prof. Daniella Vieira
daniella.vieira@unisul.br

Unidade II

Modelagem de Banco de Dados





Review

Banco de dados (BD)

Uma coleção de dados operacionais inter-relacionados e persistentes. Estes dados são gerenciados de forma independente dos programas que os utilizam, servindo assim a múltiplas aplicações de uma organização.

USUÁRIOS

SGBD

Programas de Aplicação/Consultas

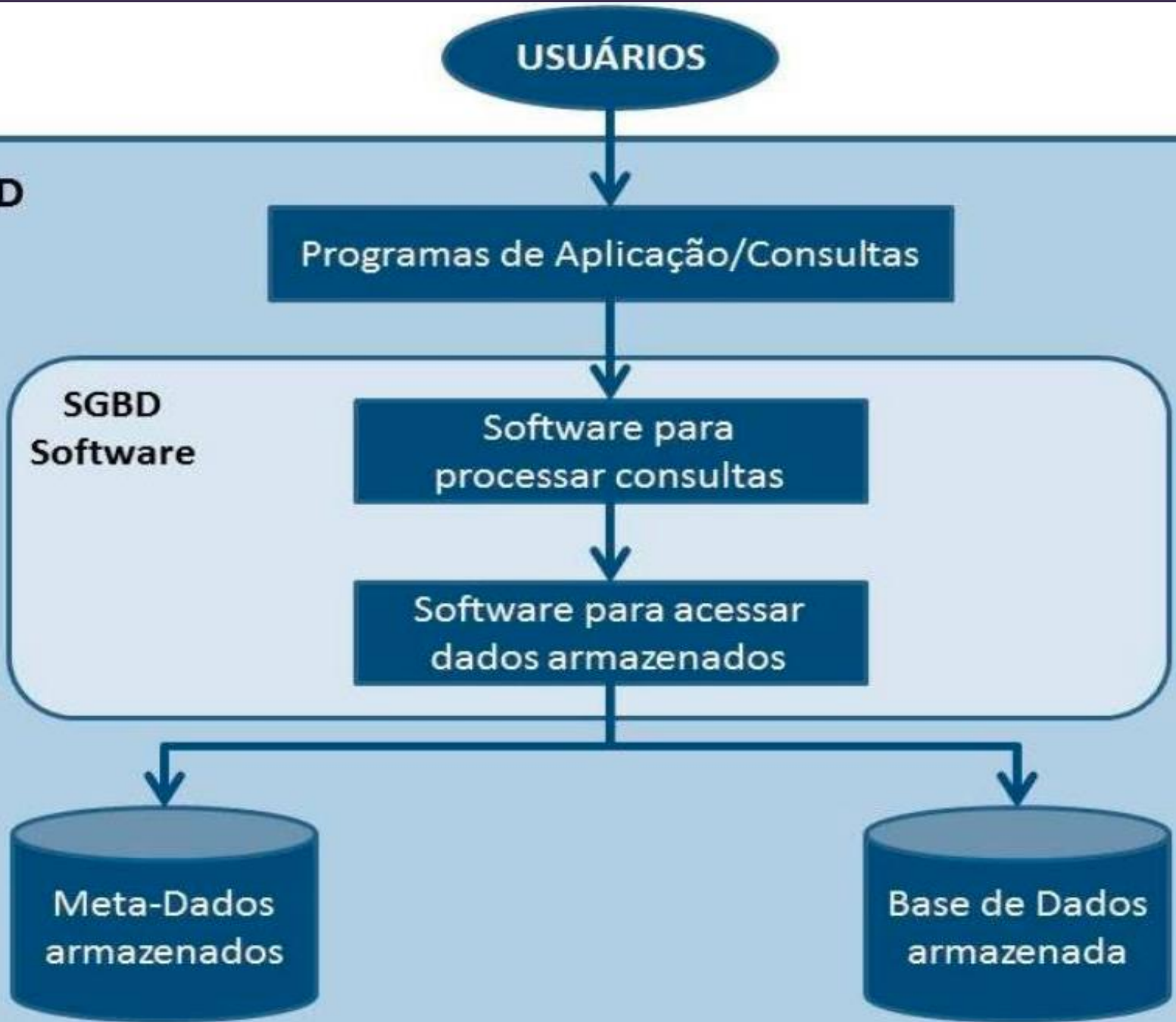
**SGBD
Software**

**Software para
processar consultas**

**Software para acessar
dados armazenados**

**Meta-Dados
armazenados**

**Base de Dados
armazenada**



Níveis de abstração

Nível físico

Esquema interno de baixo nível de abstração. Descreve como os dados estão de fato armazenados.

Descreve as estruturas de dados.

Nível lógico

Esquema conceitual de médio nível de abstração. Descreve os dados armazenados e seus relacionamentos.

Utilizado pelos administradores para definir como o banco será construído.

Nível de visão

Esquema de visão de alto nível de abstração. Proporciona diversas visões do mesmo BD.

Nível lógico

Sob a estrutura do BD está o modelo de dados. São ferramentas conceituais usadas para descrever dados, regras e relacionamentos.

Modelo lógico com base em objetos:

Utilizados na descrição de dados no nível lógico e de visões. Dispõem recursos de estruturação flexíveis

Modelo de entidade-relacionamento.

Modelo orientado a objeto.

Nível lógico

Modelo lógico com base em registros:

Utilizados na descrição de dados no nível lógico e de visões.

Especifica tanto a estrutura lógica quando a implementação de alto nível.

Modelo relacional.

Modelo de rede.

Modelo hierárquico.

Modelos de Dados

Lembrando...

Modelos de dados são formas de representação que servem para descrever as estruturas das informações contidas em um BD.

O usuário vê o banco de dados segundo um modelo de visões.

- Abordagem (Modelo) Hierárquica;
- Abordagem (Modelo) em Rede;
- Abordagem (Modelo) Relacional.

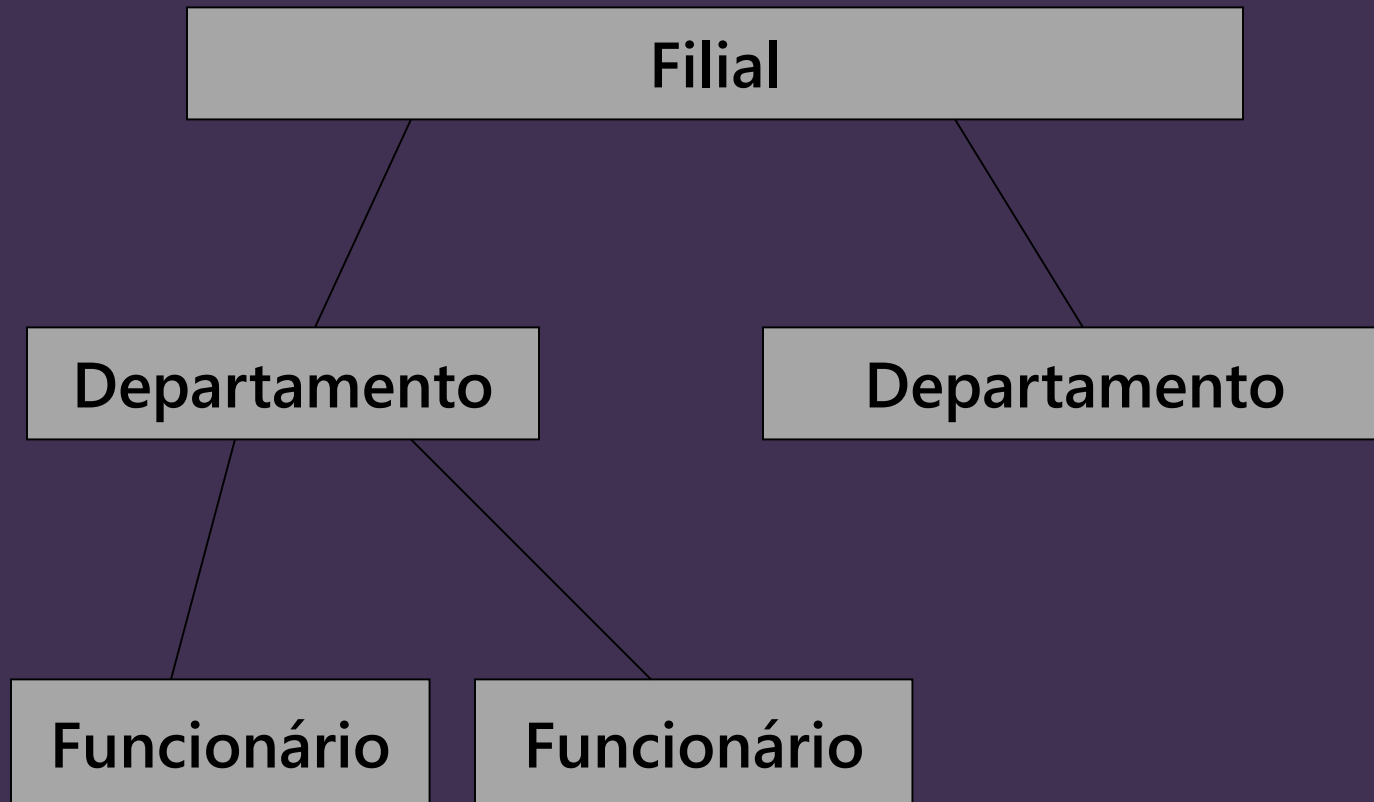
Abordagem Hierárquica

- Esta seção e as seguintes são baseadas em um banco de dados contendo as entidades: Filial, Departamento e Funcionário.
- Na abordagem hierárquica, como o próprio nome já diz, os dados são organizados de acordo com níveis hierárquicos preestabelecidos.
- Os primeiros bancos de dados estão baseados nesta abordagem. Segundo Date, “um banco de dados hierárquico, compõe-se de um conjunto ordenado de árvores – mais precisamente, de um conjunto ordenado de ocorrências múltiplas de um tipo único de árvore”.

Abordagem Hierárquica

- Na abordagem hierárquica, podemos ver o banco de dados como um único arquivo organizado em níveis. O nível superior que contém a filial é chamado de raiz.
- Qualquer acesso ao banco de dados deve ser feito a partir dele.
- Em geral, a raiz pode ter qualquer quantidade de dependentes, e estes, qualquer quantidade de dependentes de nível mais baixo.

Abordagem Hierárquica



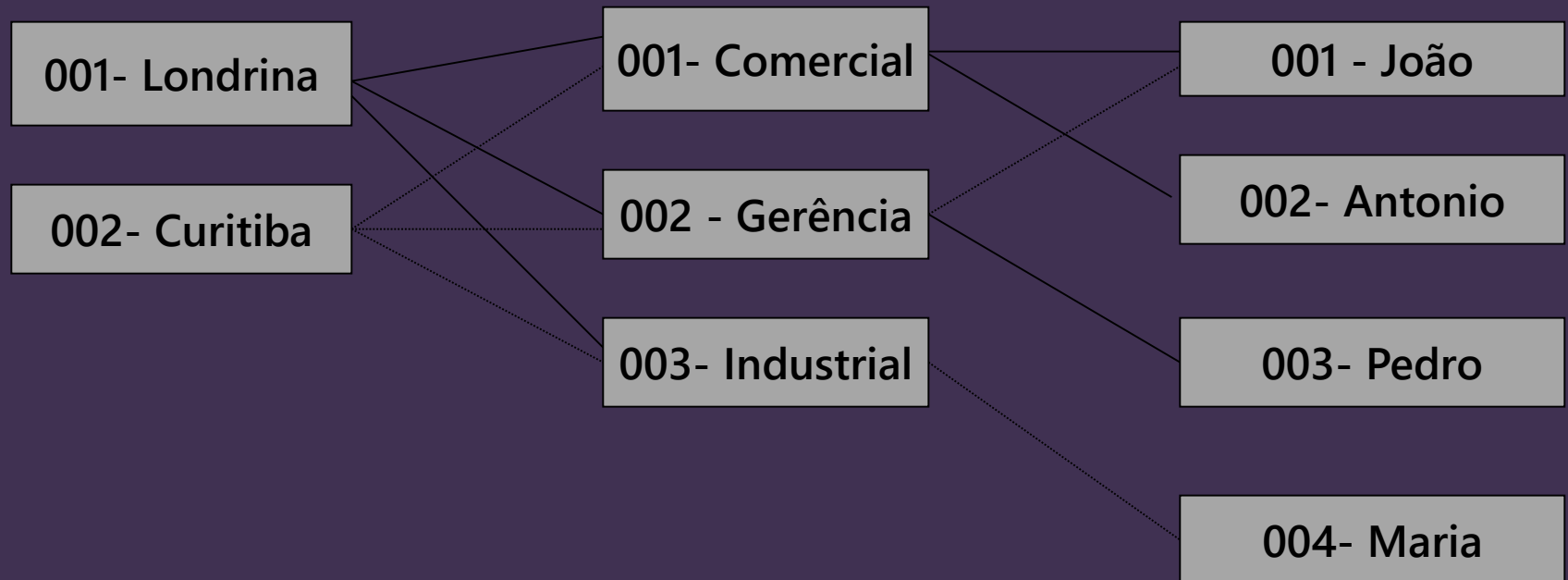
Abordagem em Rede

- No modelo em rede as informações são representadas por uma coleção de registros e o relacionamento entre elas é formado através de ligações (link).
- Extensão do modelo hierárquico.
- É uma relação membro-proprietário, na qual um membro pode ter muitos proprietários.

Abordagem em Rede

- Em um BD estruturado como um modelo em rede há freqüentemente mais de um caminho para acessar um determinado elemento de dado.
- A principal diferença entre a abordagem hierárquica e a em rede é que um registro-filho tem exatamente um pai na abordagem hierárquica, enquanto na estrutura de rede um registro-filho pode ter qualquer número de pais.

Abordagem em Rede



Abordagem em Rede

- Um banco de dados relacional consiste em uma coleção de tabelas, cada uma designada por um nome único.
- Uma tabela é uma representação bidimensional de dados composta de linhas e colunas.

Abordagem Relacional

- Padrão atual para a construção de ferramentas de BD.
- Composto de tabelas ou relações.
- Uma tabela é um conjunto não ordenado de linhas.
- Cada linha é composta por uma série de valores de campo.
- Cada campo é identificado por um nome de campo.
- O conjunto de campos das linhas de uma tabela que possuem o mesmo nome formam uma coluna.

Abordagem Relacional

Entidade:

Objeto do mundo real. Um fato.



Relacionamento:

Associação existente entre elementos de entidades



Atributo

Informações que se deseja guardar sobre o objeto

Cardinalidade

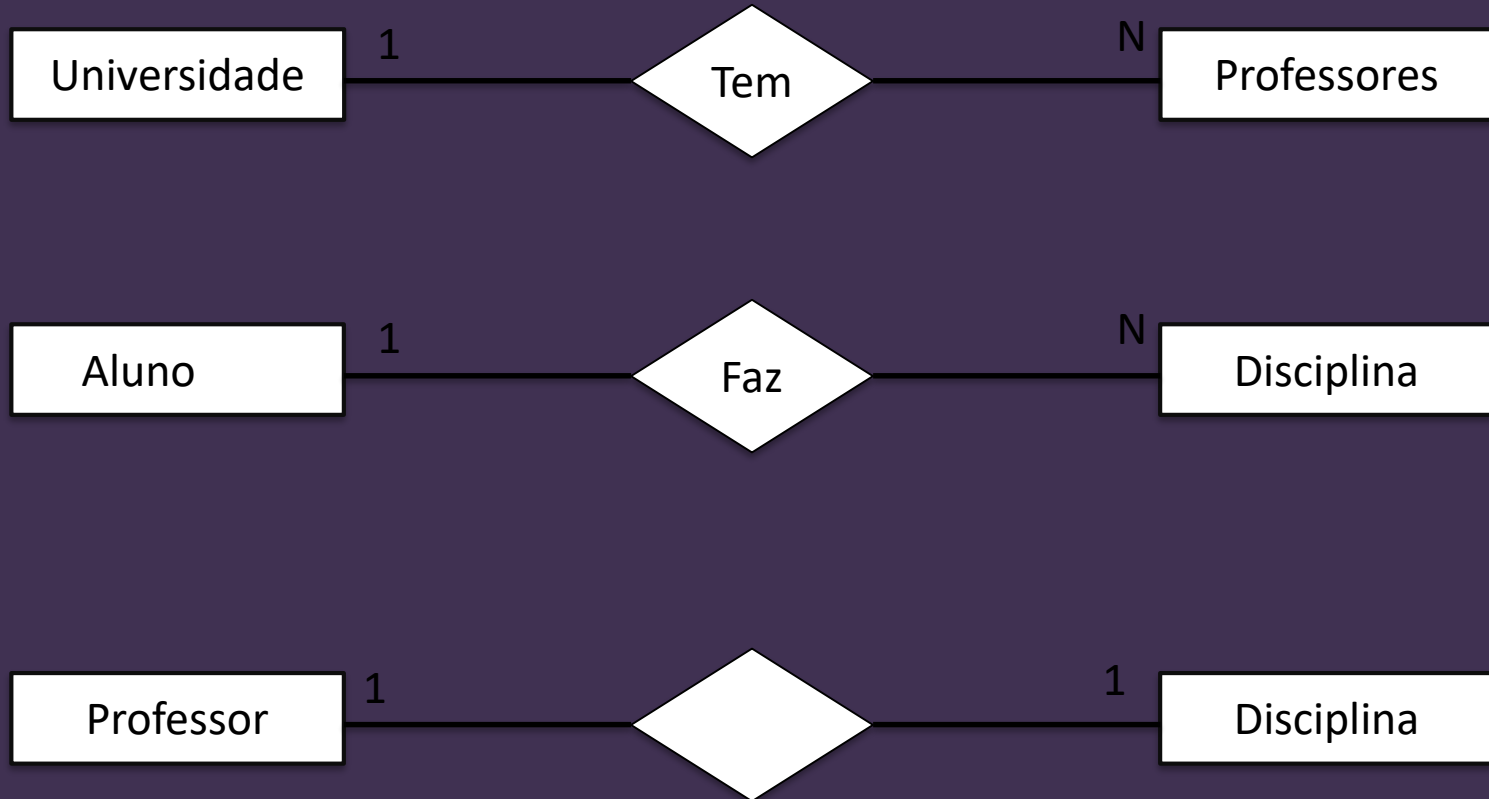
Número de ocorrências possíveis de cada entidade envolvida num relacionamento

1 .. N

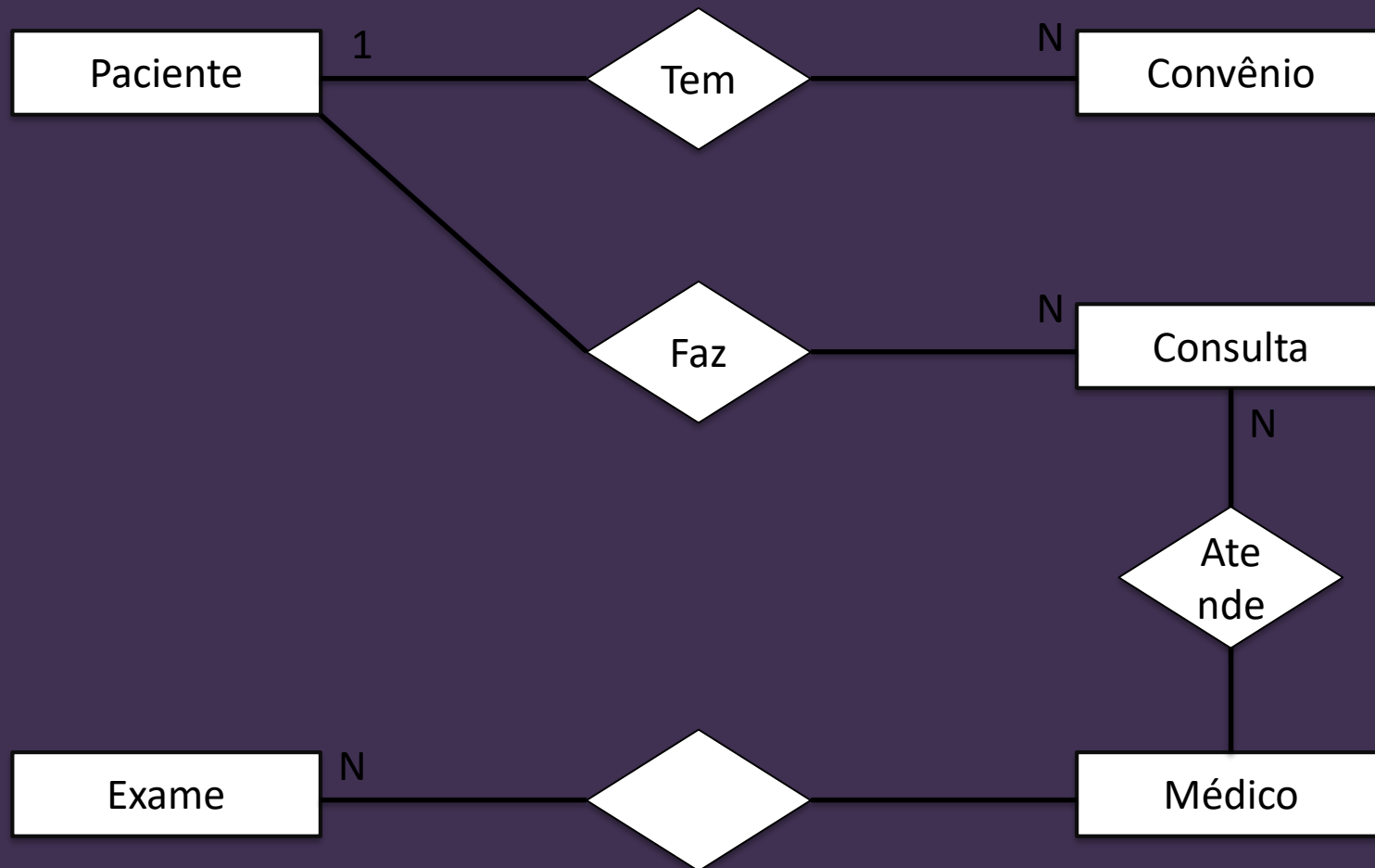
N .. M

1 .. 1

Abordagem Relacional



Abordagem Relacional

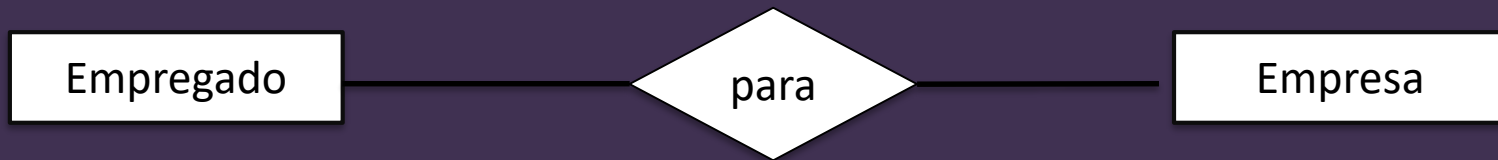




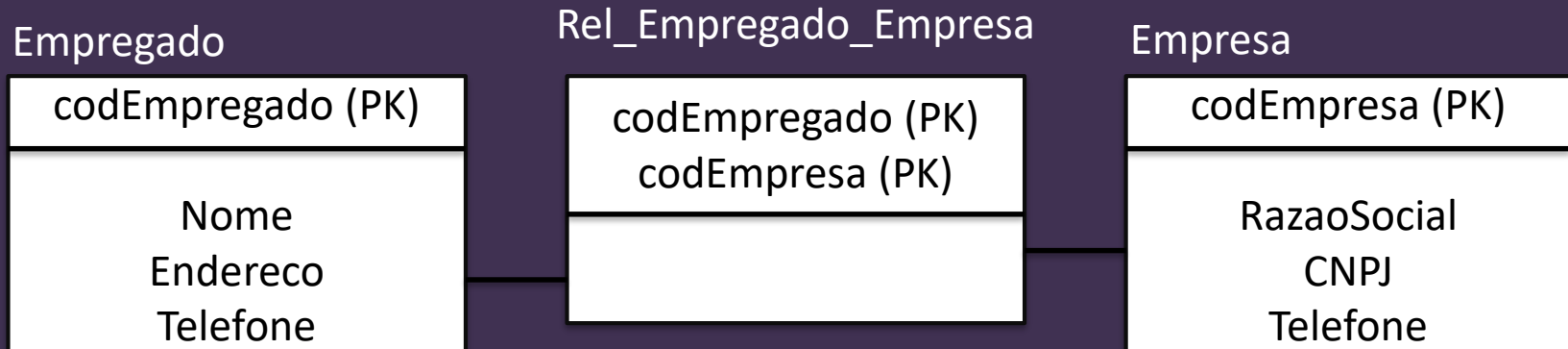
Modelagem de Dados

Modelo Conceitual vs. Modelo Lógico

Modelo conceitual

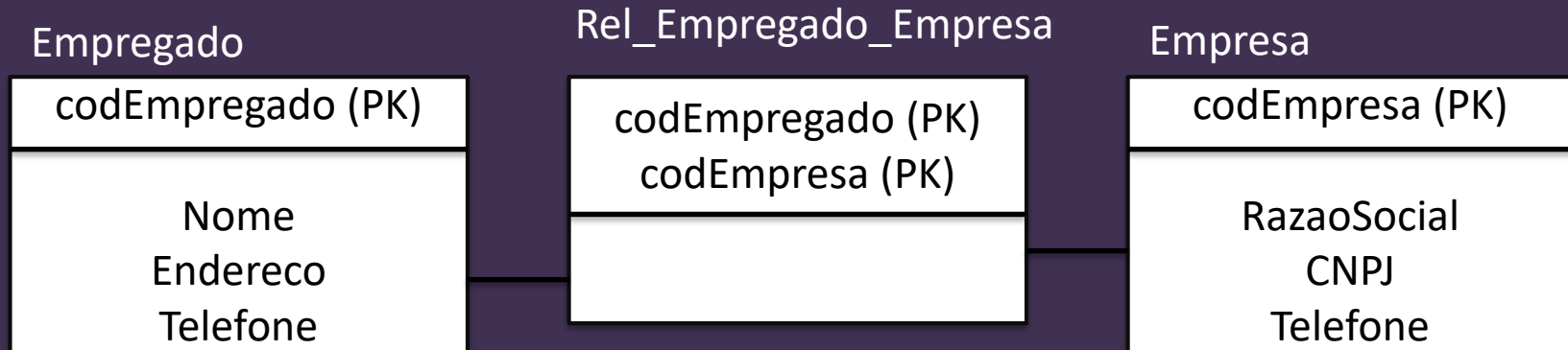


Modelo lógico



Modelo Conceitual vs. Modelo Lógico

Modelo lógico



Esquema Banco de dados

Empregado (codEmpregado, Nome, Endereco, Telefone)

Rel_Empregado_Empresa (codEmpregado, codEmpresa)

Empresa (codEmpresa, RazaoSocial, CNPJ, Telefone)

Modelo Físico

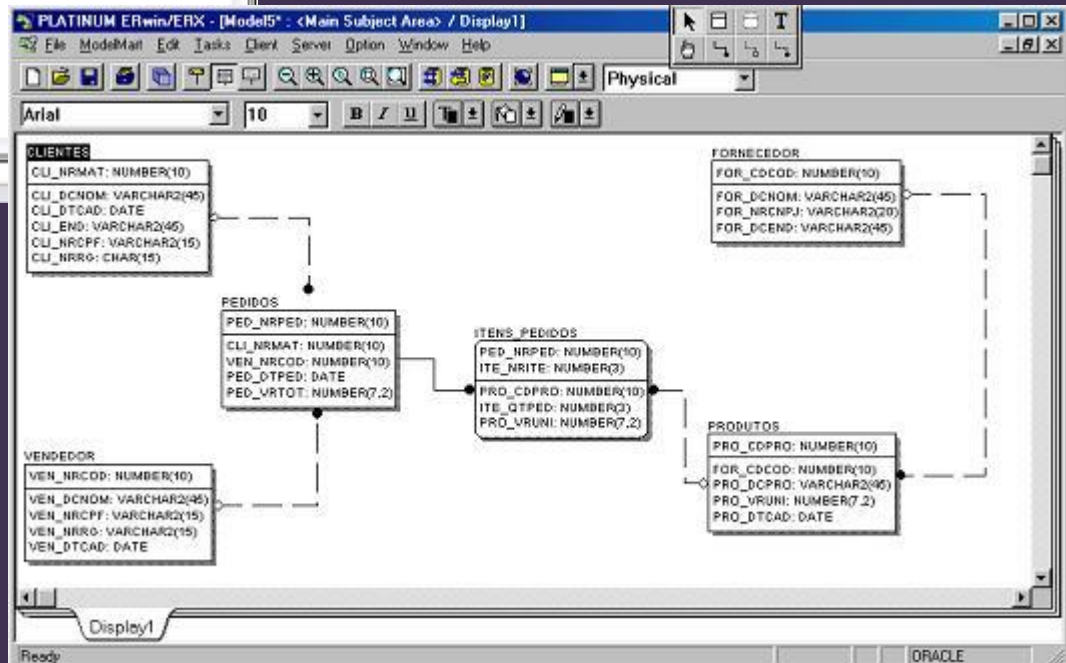
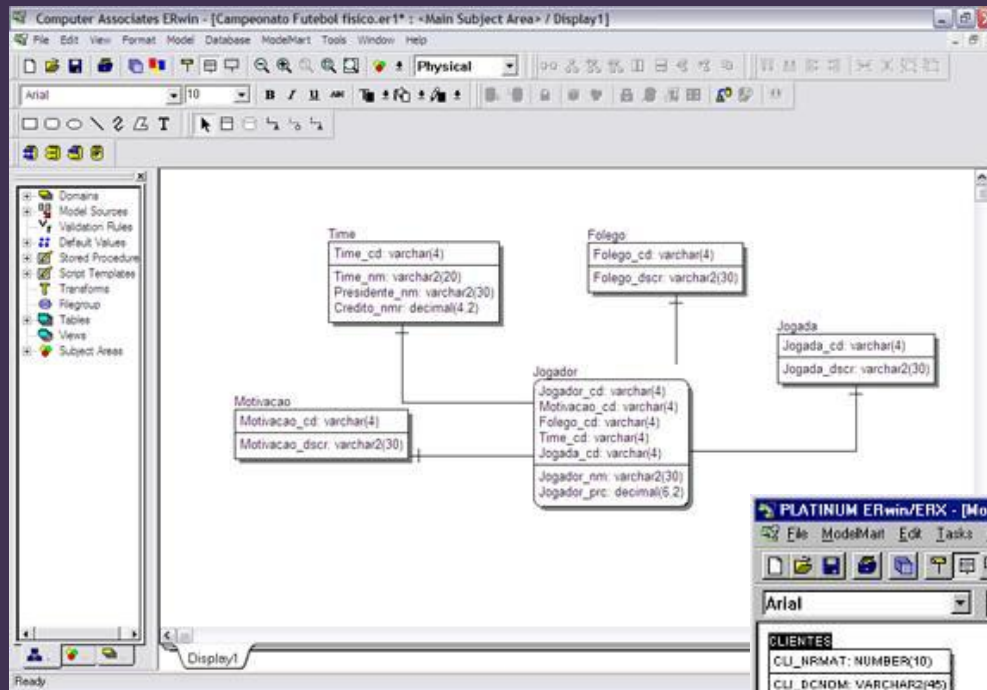
Utilizado no momento de implementar o modelo em um SGBD específico. O modelo físico contém detalhes do armazenamento interno dos dados considerando as estruturas físicas do SGBD (INT, VARCHAR, DATE, por exemplo).

pectem.en_workspace	
seq_workspace	: int(11)
nmu_plan	: varchar(20)
dta_start	: varchar(20)
dta_stop	: varchar(20)
dta_end	: varchar(20)
nmu_status	: int(11)
txt_idioma	: varchar(4)
en_contrato_seq_contrato	: int(11)
en_receivel_seq_receivel	: int(11)
en_user_seq_user	: int(11)

pectem.en_receivel	
seq_receivel	: int(11)
txt_descricao	: varchar(255)
dta_receivel	: varchar(20)
dta_pagamento	: varchar(20)
nmu_valor_bruto	: int(11)
nmu_valor_imposto	: int(11)
nmu_valor_despesa	: int(11)
nmu_valor_liquido	: int(11)

O modelo físico é utilizado por DBAs (Database Administrator).

Modelo Físico



Modelo Físico

O modelo físico contém detalhes do armazenamento interno dos dados considerando:

- índices que deverão ser criados.
- tipo da tabela (no caso do MySQL, por exemplo, podemos optar por tabelas do tipo MyISAM, InnoDB).
- detalhes de armazenamento como TABLESPACES (por exemplo, em Oracle).

Visualizar Estrutura SQL Procurar Inserir Exportar Importar Operações Limpar Eliminar									
	Campo	Tipo	Collation	Atributos	Nulo	Padrão	Extra	Ação	
<input type="checkbox"/>	seq_atividade	int(11)			Não	None	AUTO_INCREMENT		
<input type="checkbox"/>	txt_titulo	varchar(255)	utf8_general_ci		Não	None			
<input type="checkbox"/>	txt_descricao	text	utf8_general_ci		Sim	NULL			
<input type="checkbox"/>	dta_inicio	varchar(20)	utf8_general_ci		Sim	NULL			
<input type="checkbox"/>	dta_conclusao	varchar(20)	utf8_general_ci		Sim	NULL			
<input type="checkbox"/>	dta_entrega	varchar(20)	utf8_general_ci		Não	None			
<input type="checkbox"/>	vl_parcial	varchar(11)	utf8_general_ci		Sim	NULL			
<input type="checkbox"/>	hr_planejada	int(11)			Sim	NULL			
<input type="checkbox"/>	hr_realizada	int(11)			Sim	NULL			
<input type="checkbox"/>	nmu_prioridade	int(11)			Sim	NULL			
<input type="checkbox"/>	nmu_status	int(11)			Sim	NULL			

Modelo Físico

Os detalhes do armazenamento não têm influência direta na programação da aplicação que irá acessar o SGBD. No entanto, influenciam diretamente a performance geral do banco de dados melhorando ou piorando o tempo de resposta de uma consulta, por exemplo.

A estrutura de dados das colunas pode ter influência sobre a performance das aplicações que acessarão o SGBD.

Modelo Físico

- Boas práticas do Oracle:

Armazenar os índices e as tabelas em TABLESPACES diferentes, para aumentar a performance geral do banco de dados.

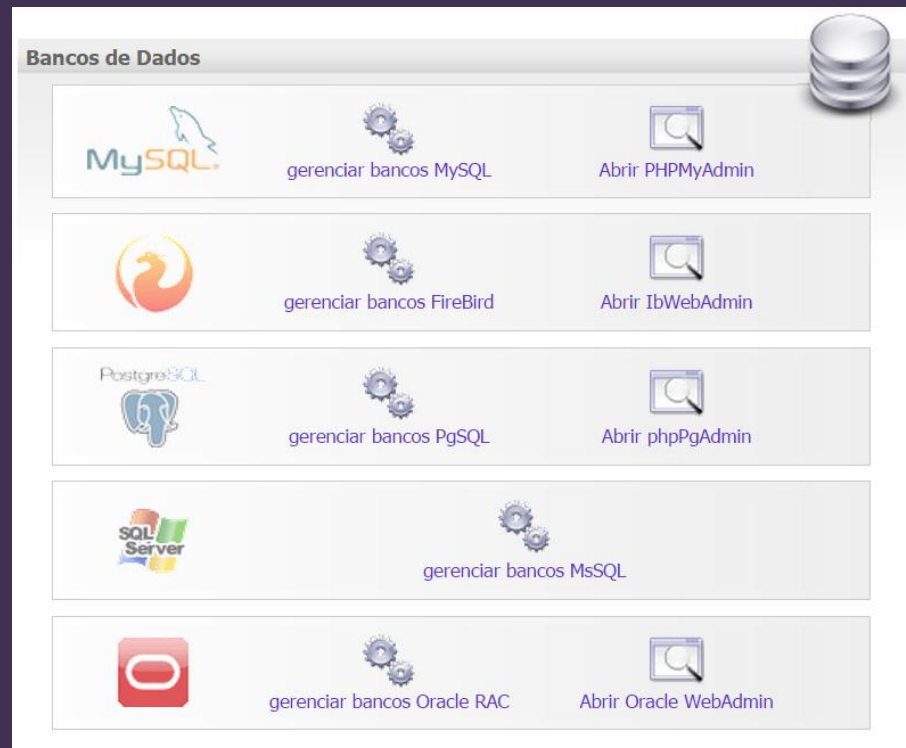
- Boas práticas do MySQL:

O tipo de tabela MyISAM não foi desenhada para sistemas transacionais, ou OLTP. Para sistemas puramente transacionais o aconselhável é a utilização de tabelas do tipo InnoDB.

Estrutura												SQL	Procurar	Procurar por exemplo	Exportar	Importar	Designer	Operações
	Tabela ▲		Ação					Registros ¹	Tipo	Collation	Tamanho	Sobrecarga						
<input type="checkbox"/>	en_atividade							14	InnoDB	utf8_general_ci	16.0 KB	-						
<input type="checkbox"/>	en_atividade_has_en_equipe							0	InnoDB	utf8_general_ci	16.0 KB	-						
<input type="checkbox"/>	en_categoria							4	InnoDB	utf8_general_ci	16.0 KB	-						
<input type="checkbox"/>	en_cliente							5	InnoDB	utf8_general_ci	16.0 KB	-						

Modelo Físico

O modelo físico é totalmente dependente do SGBD escolhido. A maneira com que o otimizador Oracle trata a estratégia de acesso aos dados é diferente do SGBD DB2, por exemplo.



Modelo Físico

Utilização total de SQL (Structure Query Language).

```
CREATE TABLE IF NOT EXISTS `en_atividade` (  
  `seq_atividade` int(11) NOT NULL AUTO_INCREMENT,  
  `txt_titulo` varchar(255) NOT NULL,  
  `txt_descricao` text,  
  PRIMARY KEY (`seq_atividade`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1;
```

Campo	Tipo	Collation	Atributos	Nulo	Padrão	Extra
<u>seq_atividade</u>	int(11)			Não	<i>None</i>	AUTO_INCREMENT
txt_titulo	varchar(255)	utf8_general_ci		Não	<i>None</i>	
txt_descricao	text	utf8_general_ci		Sim	<i>NULL</i>	



SQL

Structure Query Language

- Entre 1974 e 1979, o San José Research Laboratory da IBM desenvolveu um SGDB relacional que ficou conhecido como Sistema R. Para a criação e acesso aos dados foi adotada uma linguagem chamada SEQUEL, mais tarde rebatizada SQL (*Structured Query Language*).
- Adotada como padrão mundial pela ISO em 1987, SQL é uma linguagem exclusiva de banco de dados Cliente/Servidor.
- Embora a *query* em sua definição, a SQL foi projetada de forma a permitir que além de consultas (queries), inserções, alterações e deleções fossem feitas, além da própria criação das tabelas e campos.
- Dividiu-se a SQL então em duas partes: DDL (*Data Description Language*); e DML (*Data Manipulation Language*).

Structure Query Language

- A DDL, uma parte muito pequena da SQL, permite a criação e manutenção do dicionário de dados. O dicionário de dados contém a definição de cada tabela, de cada campo, enfim, contém a definição da base de dados propriamente dita. Em outras palavras, o dicionário de dados guarda dados sobre os dados.
- Embora existam algumas outras construções, a mais importante das construções da DDL é a destinada a criação de tabelas.

```
CREATE TABLE COMPANHIA (  
    CODIGO INTEGER,  
    NOME CHAR (40),  
    SEDE CHAR (50) ),  
PRIMARY KEY CODIGO;
```

Structure Query Language

- Já a DML é a parte mais ampla da SQL. Permite pesquisar, alterar, incluir e deletar dados da base de dados. São quatro as sentenças mais importantes da DML:
 - SELECT: permite a pesquisa de dados;
 - UPDATE: permite a atualização de dados;
 - DELETE: permite a exclusão de dados;
 - INSERT: permite a inclusão de dados.

Structure Query Language

Indique no modelo:

- Chave primária
- Chave estrangeira
- Tupla
- Atributo
- Tabela

COMPANHIA				
CODIGO	NOME	AERONAVES	SEDE	PRESIDENTE
JJ	TAM	500	SPO	YIX
AV	AVIANCA	120	SPO	IJH
GG	GOL	450	SPO	AKS

VOOS				
VOO	TARIFA	ASSENTOS	CLASSE	COMPANHIA
JJ3021	100,00	302	ECONOMICA	TAM
JJ 3099	500,00	150	ECONOMICA	TAM
GG 4000	600,00	302	ECONOMICA	GOL

PASSAGEIRO				
ID	DOCUMENTO	NOME	IDADE	SEXO
1	12.312.312	Fulano	34	M
2	664.234	Beltrano	23	M
3	234.456.346	Ciclano	15	M

Structure Query Language

- DML para selecionar a informação de quantos voos possuem valores inferiores a 1000,00.

```
SELECT VOO, TARIFA, ASSENTOS  
FROM VOOS  
WHERE TARIFA < 1000,00;
```

- Obter para os voos cadastrados o nome da Cia e a Sede.

```
SELECT VOO, TARIFA, NOME, SEDE  
FROM VOOS, COMPANHIA  
WHERE NOME.COMPANHIA = VOOS.COMPANHIA;
```

Structure Query Language

- Para obter os voos cadastrados com tarifa < 1000,00 o nome da Cia e a Sede.

```
SELECT VOO, TARIFA, NOME, SEDE
FROM VOOS, COMPANHIA
WHERE COMPANHIA.NOME = VOOS.NOME
      AND VOOS.TARIFA < 1000,00;
```

- Para saber quem viajou, quando e por qual companhia?

```
SELECT <COLUNAS>
FROM <RELAÇÃO>
WHERE <CONDIÇÃO>;
```

Structure Query Language

- Para inserir dados?

```
INSERT INTO <tabela>  
    [<campos>]  
    [VALUES <valores> ]
```

- Para atualizar dados?

```
UPDATE <tabela> SET  
    <campos> = <valores>  
    WHERE <condicao>;
```

- Para excluir dados?

```
DELETE FROM <tabela> WHERE <condicao>;
```

Comparativo dos modelos

Modelagem Conceitual

- Diagrama Entidade e Relacionamento (DER).
- O modelo básico do BD.
- Visão macro.

Modelagem Lógica

- Relaciona as características e restrições do modelo conceitual com as do modelo selecionado para implementação.
- São definidos os padrões e nomenclaturas, chaves primárias e estrangeiras.

Modelagem Física

- Nível mais baixo de abstração.

Comparativo dos modelos

Modelo	Perfil	Grau de abstração	Foco	Independência
Conceitual	Arquiteto de BD	Média-Alta	Visão global dos dados (independe do modelo de BD)	Hardware e Software
Lógica	Arquiteto de BD	Média-Baixa	Modelo específico do BD	Hardware
Físico	DBA	Baixo	Método de armazenamento e acesso	Dependente

Atividade



Atividade

- A manipulação de uma BD envolve consultas, atualizações, inclusões e exclusões.

Inserir <'João da Silva', 10, 1, MAT> em ALUNO

Excluir a tupla de ALUNO onde Numero_aluno=17

Alterar o Nome em ALUNO com Curso = MAT para 'André'

Selecionar ALUNO onde Curso = CC

- As operações sobre o banco precisam ser especificadas na linguagem do SGBD antes de serem processadas.

- 1) Elabore o modelo lógico e físico para este sistema
- 2) Construa o banco utilizando a linguagem SQL

Referências bibliográficas

DATE, C. J. **Introdução a sistemas de bancos de dados**. Rio de Janeiro: Campus, 2004. 865 p. ISBN 85-352-1273-6.

ELMASRI, Ramez; NAVATHE, Sham. **Sistemas de banco de dados**. 4. ed. São Paulo: Addison-Wesley, 2005. 724 p. ISBN 8588639173.

NAVATHE, Elmasri. **Sistema de Banco de Dados**. 6ª Edição. São Paulo: Person Addison Wesley, 2011.

PEREIRA, Silvio do Lago. **Estruturas de dados fundamentais: conceitos e aplicações**. 8. ed. São Paulo: Érica, 2004. 238 p. ISBN 85-7194-370-2.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Sistema de banco de dados**. 3. ed. São Paulo: Makron Books, 2004. 778 p. ISBN 85-346-1073-8.

Material complementar



Oracle Academy

- [DFo_3_1_pr.pdf](#)
- [DFo_3_2_pr.pdf](#)
- [DFo_3_4_pr.pdf](#)
- [DFo_4_1_pr.pdf](#)
- [DFo_4_2_pr.pdf](#)