

GRADUAÇÃO EM  
**SISTEMAS  
DE INFORMAÇÃO**

**Banco de Dados I**  
Aula 04

Prof. MSc. Érick de Souza Carvalho

1

**Banco de Dados I**  
Aula 04  
Prof. MSc. Érick de Souza Carvalho

## Revisão das Aulas Anteriores

### 😊 Aula 01

- 😊 Definição e a relação entre: BD X Sist. BD X SGBD

### 😊 Aula 02

- 😊 Sist. Arquivos e suas dificuldades, desvantagens e problemas

### 😊 Aula 03

- 😊 Soluções oferecidas pelos B.D. para estes problemas

## Revisão das Aulas Anteriores

### 😊 Aula 01

- 😊 Definição e a relação entre: BD X Sist. BD X SGBD

### 😊 Aula 02

- 😊 Sist. Arquivos e suas dificuldades, desvantagens e problemas

### 😊 Aula 03

- 😊 Soluções oferecidas pelos B.D. para estes problemas

## Esquema X Instância

- Semelhantes aos tipos e variáveis nas linguagens de programação
- **Esquema** – a estrutura lógica do banco de dados
  - Exemplo: O banco de dados consiste nas informações sobre um conjunto de clientes e contas e na relação entre eles
  - Análogo às informações de tipo de uma variável em um programa
  - Esquema físico: projeto de banco de dados no nível físico
  - Esquema lógico: projeto de banco de dados no nível lógico
- **Instância** – o conteúdo real do banco de dados em um determinado ponto no tempo
  - Análogo ao valor de uma variável

### Esquema X Instância

- Semelhantes aos tipos e variáveis nas linguagens de programação
- **Esquema** – a estrutura lógica do banco de dados
  - Exemplo: O banco de dados consiste nas informações sobre um conjunto de clientes e contas e na relação entre eles
  - Análogo às informações de tipo de uma variável em um programa
  - Esquema físico: projeto de banco de dados no nível físico
  - Esquema lógico: projeto de banco de dados no nível lógico
- **Instância** – o conteúdo real do banco de dados em um determinado ponto no tempo
  - Análogo ao valor de uma variável

## Abstração de Dados

É uma das principais finalidades de um Bancos de Dados.

É uma visão abstrata dos dados.

Permite acessarmos os dados independente de como eles estão armazenados.

### Níveis de Abstração:

**Nível Físico** – Como estão fisicamente no(s) Servidor (es). Nível Baixo de Abstração.

**Nível Lógico** – Quais dados estão armazenados e a relação entre eles. Nível Alto de Abstração.

**Nível de Visão(View)** – Os dados trabalhados, para simplificar ou ocultar certas informações. Podemos ter várias Visões dos mesmos dados. É o Nível mais Alto de Abstração.

## Abstração de Dados

É uma das principais finalidades de um Bancos de Dados.

É uma visão abstrata dos dados.

Permite acessarmos os dados independente de como eles estão armazenados.

### Níveis de Abstração:

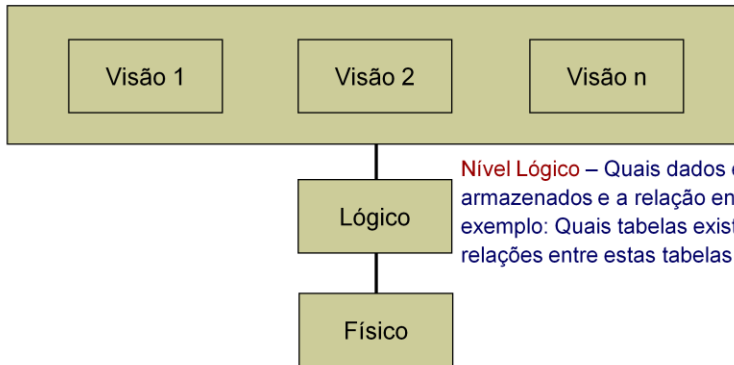
**Nível Físico** – Como estão fisicamente no(s) Servidor (es). Nível Baixo de Abstração.

**Nível Lógico** – Quais dados estão armazenados e a relação entre eles. Nível Alto de Abstração.

**Nível de Visão(View)** – Os dados trabalhados, para simplificar ou ocultar certas informações. Podemos ter várias Visões dos mesmos dados. É o Nível mais Alto de Abstração.

## Níveis de Abstração de Dados

**Nível de Visão** – Criamos várias visões dos dados, para: trabalhar, ocultar ou simplificar certas informações, por exemplo: criamos views sobre as tabelas



**Nível Lógico** – Quais dados estão armazenados e a relação entre eles, por exemplo: Quais tabelas existem, as relações entre estas tabelas e seus campos

**Nível Físico** – Como o banco de dados está fisicamente no Servidor, por exemplo: Em quais discos e em quais arquivos o BD está distribuído

## Níveis de Abstração de Dados

**Nível de Visão** – Criamos várias visões dos dados, para: trabalhar, ocultar ou simplificar certas informações, por exemplo: criamos views sobre as tabelas

**Nível Lógico** – Quais dados estão armazenados e a relação entre eles, por exemplo: Quais tabelas existem, as relações entre estas tabelas e seus campos

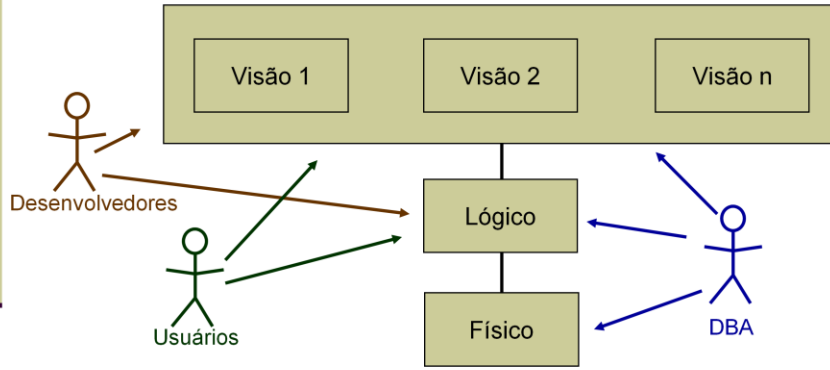
**Nível Físico** – Como o banco de dados está fisicamente no Servidor, por exemplo: Em quais discos e em quais arquivos o BD está distribuído

## Níveis de Abstração de Dados (*Quem acessa*)

**Desenvolvedores:** Criam objetos e inserem dados nos níveis: de visão e Lógico

**Usuários:** Inserem dados nos níveis: de visão e Lógico

**DBA:** Cuida do nível físico, Cria objetos e insere dados nos níveis: de visão e Lógico



- Podemos entender criar objetos como: Criar, Alterar e Excluir Objetos
- Entendamos inserir dados como: Inserir, alterar, excluir e consultar dados

## Níveis de Abstração de Dados (*Quem acessa*)

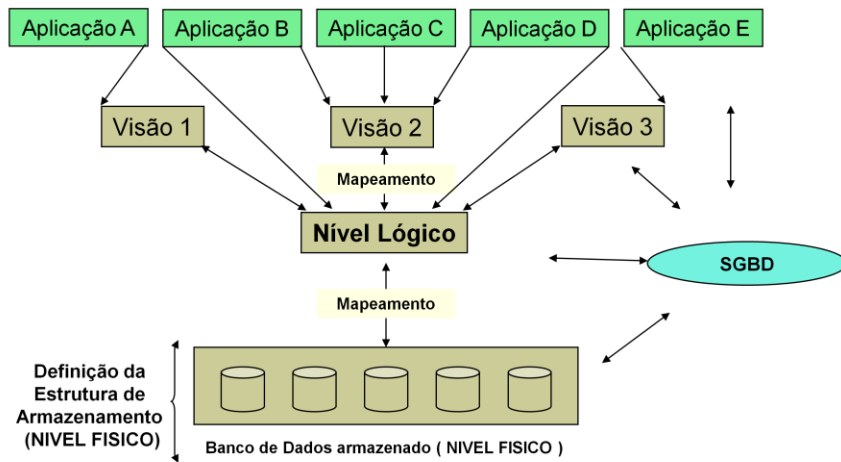
**Desenvolvedores:** Criam objetos e inserem dados nos níveis: de visão e Lógico

**Usuários:** Inserem dados nos níveis: de visão e Lógico

**DBA:** Cuida do nível físico, Cria objetos e insere dados nos níveis: de visão e Lógico

- Podemos entender criar objetos como: Criar, Alterar e Excluir Objetos
- Entendamos inserir dados como: Inserir, alterar, excluir e consultar dados

## Níveis de Abstração de Dados (Continuação)



### Níveis de Abstração de Dados (Continuação)

As aplicações (Sistemas de Banco de Dados) acessam os níveis lógico e de visão.

O SGBD acessa todos os níveis.

## Independência de Dados Física

- **Independência de dados física** – a capacidade de modificar o nível de abstração físico sem mudar o nível de abstração lógico
  - Aplicações dependem do esquema lógico, com isso, quando mudamos o nível físico as aplicações não são afetadas
  - Em geral, as interfaces entre os vários níveis e componentes devem ser bem-definidas de modo que as mudanças em algumas partes não influenciem seriamente outras
  - Muitas vezes o DBA faz mudanças no nível físico sem que usuários das aplicações e desenvolvedores percebam

### Independência de Dados Física

■ **Independência de dados física** – a capacidade de modificar o nível de abstração físico sem mudar o nível de abstração lógico

- Aplicações dependem do esquema lógico, com isso, quando mudamos o nível físico as aplicações não são afetadas
- Em geral, as interfaces entre os vários níveis e componentes devem ser bem-definidas de modo que as mudanças em algumas partes não influenciem seriamente outras
- Muitas vezes o DBA faz mudanças no nível físico sem que usuários das aplicações e desenvolvedores percebam



## Tarefas do DBA (Administrador de Banco de Dados)

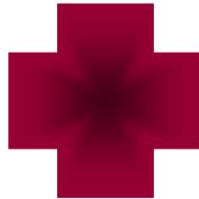
✓ Gerenciar a disponibilidade dos bancos de dados

✓ Planejar e criar Bancos de dados

✓ Gerenciar estruturas físicas

✓ Gerenciar armazenamento segundo o projeto

✓ Gerenciar Segurança



✓ Administração de Rede

✓ Backup e Recovery

✓ Performance e Tuning

### Tarefas de um DBA

- Gerenciar a disponibilidade dos bancos de dados
- Instalar e administrar os SGBDs e os servidores de banco de dados
- Planejar e criar bancos de dados e instâncias
- Gerenciar estruturas físicas
- Gerenciar estruturas lógicas
- Gerenciar armazenamento segundo o projeto
- Gerenciar Segurança
- Administração de Rede
- Backup e Recovery
- Performance e Tuning

# Evolução dos Bancos de Dados



UniFOA – Centro Universitário de Volta Redonda

Prof. MSc. Érick de S. Carvalho

BD1\_3P\_Aula04A 10

## Evolução dos Bancos de Dados

- O primeiro banco de dados foi o Hierárquico, foi uma verdadeira revolução, foi utilizado pela NASA para mandar o Homem a Lua, era mais adequado para informações naturalmente no formato hierárquico, como por exemplo estrutura organizacional de uma empresa, porém era inadequado para outros tipos de informações, por não ser muito flexível, exigindo assim um esforço muito grande para este tipo de sistema. Um registro filho poderia ter apenas um registro pai e os ponteiros entre os registros eram físicos;
- O Banco Modelo de Redes foi uma evolução do Hierárquico, porém um filho poderia ter vários pais. Era muito mais flexível que o hierárquico, possibilitando a implantação de muito mais formatos de informações, porém a medida que os ponteiros entre os registros crescia, ficava cada vez mais complexa sua administração. Embora complexo, muitos dos sistemas de informações relacionais também poderiam ser implementados como modelo de redes;
- Relacional - Foi outra grande revolução, onde os dados eram agrupados seguindo a teoria dos conjuntos, foram criados os conceitos de Entidades, Relacionamentos e Atributos, porém estes relacionamentos eram lógicos;
- Orientados a Objetos, ou Nativamente Orientado a Objetos, surgiram com base na programação orientada a objetos, implementam no banco de dados os conceitos da Orientação a Objetos, como Hierarquia, Herança, Encapsulamento etc... É bem melhor que o relacional no tratamento de objetos grandes e objetos multimídia;
- Objeto-Relacional - É o banco de dados relacional com as features do Orientado a Objetos, a maioria dos Bancos de Dados que estão no mercado hoje são deste tipo, como: SQL Server e Oracle;
- Orientado a Aspectos - É uma evolução do Orientado a Objetos e assim como ele é baseado na programação Orientada a Aspectos. Ainda não existe, porém segundo os acadêmicos a expectativa é que surgirá em alguns anos.

## Evolução dos Bancos de Dados (*Hierárquico*)

### Hierárquico:

O primeiro banco de dados que surgiu foi o Hierárquico;

Foi uma verdadeira revolução;

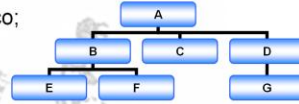
Foi utilizado pela NASA para mandar o Homem a Lua;

Era mais adequado para informações naturalmente no formato hierárquico, como por exemplo estrutura organizacional de uma empresa;

Era inadequado para outros tipos de informações, por não ser muito flexível, exigindo assim um esforço muito grande para este tipo de sistema; e

Um registro filho poderia ter apenas um registro pai (*esta era a sua principal limitação*);

Os ponteiros entre os registros eram físicos (*Esta era outra limitação importante*).



### Hierárquico:

O primeiro banco de dados que surgiu foi o Hierárquico;

Foi uma verdadeira revolução;

Foi utilizado pela NASA para mandar o Homem a Lua;

Era mais adequado para informações naturalmente no formato hierárquico, como por exemplo estrutura organizacional de uma empresa;

Era inadequado para outros tipos de informações, por não ser muito flexível, exigindo assim um esforço muito grande para este tipo de sistema; e

Um registro filho poderia ter apenas um registro pai (*esta era a sua principal limitação*);

Os ponteiros entre os registros eram físicos (*Esta era outra limitação importante*).

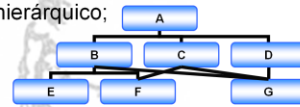
## Evolução dos Bancos de Dados (De Redes)

### De Redes:

Não está relacionado com redes de computadores e sim com redes de relacionamentos;

O banco de dados modelo de redes foi uma evolução do hierárquico;

Um registro filho poderia ter vários pais;



Era mais flexível que o hierárquico, possibilitando a implantação de mais formatos de informações;

*Os ponteiros entre os registros continuavam sendo físicos; (Esta era sua principal limitação); e*

*À medida que a quantidade de ponteiros entre os registros crescia, ficava cada vez mais complexa a sua administração.*

***“Embora complexo, muitos dos sistemas de informação implementados usando bancos de dados relacionais poderiam tranquilamente usar o modelo de redes.”***

### De Redes:

Não está relacionado com redes de computadores e sim com redes de relacionamentos;

O banco de dados modelo de redes foi uma evolução do hierárquico;

Um registro filho poderia ter vários pais;

Era mais flexível que o hierárquico, possibilitando a implantação de mais formatos de informações;

*Os ponteiros entre os registros continuavam sendo físicos; (Esta era sua principal limitação); e*

*À medida que a quantidade de ponteiros entre os registros crescia, ficava cada vez mais complexa a sua administração.*

***“Embora complexo, muitos dos sistemas de informação implementados usando bancos de dados relacionais poderiam tranquilamente usar o modelo de redes.”***

## Evolução dos Bancos de Dados (*Relacional e OO*)

### Relacional:

Foi outra grande revolução;

Os dados eram agrupados seguindo a teoria dos conjuntos;

Foram criados os conceitos de Entidades, Relacionamentos e Atributos;

Os ponteiros entre os registros eram lógicos; e

Estes ponteiros eram chamados de relacionamentos.

### Orientados a Objetos (OO – *Object Oriented*):

Também chamados de “Nativamente Orientado a Objetos”;

Surgiram com base na programação orientada a objetos;

Implementam no banco de dados os conceitos da Orientação a Objetos, como Hierarquia, Herança, Encapsulamento etc...

### Relacional:

Foi outra grande revolução;

Os dados eram agrupados seguindo a teoria dos conjuntos;

Foram criados os conceitos de Entidades, Relacionamentos e Atributos;

Os ponteiros entre os registros eram lógicos; e

Estes ponteiros eram chamados de relacionamentos.

### Orientados a Objetos (OO – *Object Oriented*):

Também chamados de “Nativamente Orientado a Objetos”;

Surgiram com base na programação orientada a objetos;

Implementam no banco de dados os conceitos da Orientação a Objetos, como Hierarquia, Herança, Encapsulamento etc...

## Evolução dos Bancos de Dados (*Objeto-Relacional e OA*)

### Objeto-relacional:

Também chamado de “Relacional de Objeto”;

É o banco de dados relacional acrescido das features do Orientado a Objetos; e

A maioria dos grandes bancos de dados que estão no mercado hoje são deste tipo, como: SQL Server e Oracle;

### Orientados a Aspectos (AO – *Aspect Oriented*):

Ainda não existe de fato, porém segundo os acadêmicos a expectativa é que surgirá em alguns anos;

É uma provável evolução do Orientado a Objetos;

Assim como o OO foi baseado na programação orientada a objetos, o AO deverá ser baseado na orientação a aspectos;

Deverá no banco de dados os conceitos de orientação a aspectos.

### Objeto-relacional:

Também chamado de “Relacional de Objeto”;

É o banco de dados relacional acrescido das features do Orientado a Objetos; e

A maioria dos grandes bancos de dados que estão no mercado hoje são deste tipo, como: SQL Server e Oracle;

### Orientados a Aspectos (AO – *Aspect Oriented*):

Ainda não existe de fato, porém segundo os acadêmicos a expectativa é que surgirá em alguns anos;

É uma provável evolução do Orientado a Objetos;

Assim como o OO foi baseado na programação orientada a objetos, o AO deverá ser baseado na orientação a aspectos;

Deverá no banco de dados os conceitos de orientação a aspectos.