

# Encontro 13

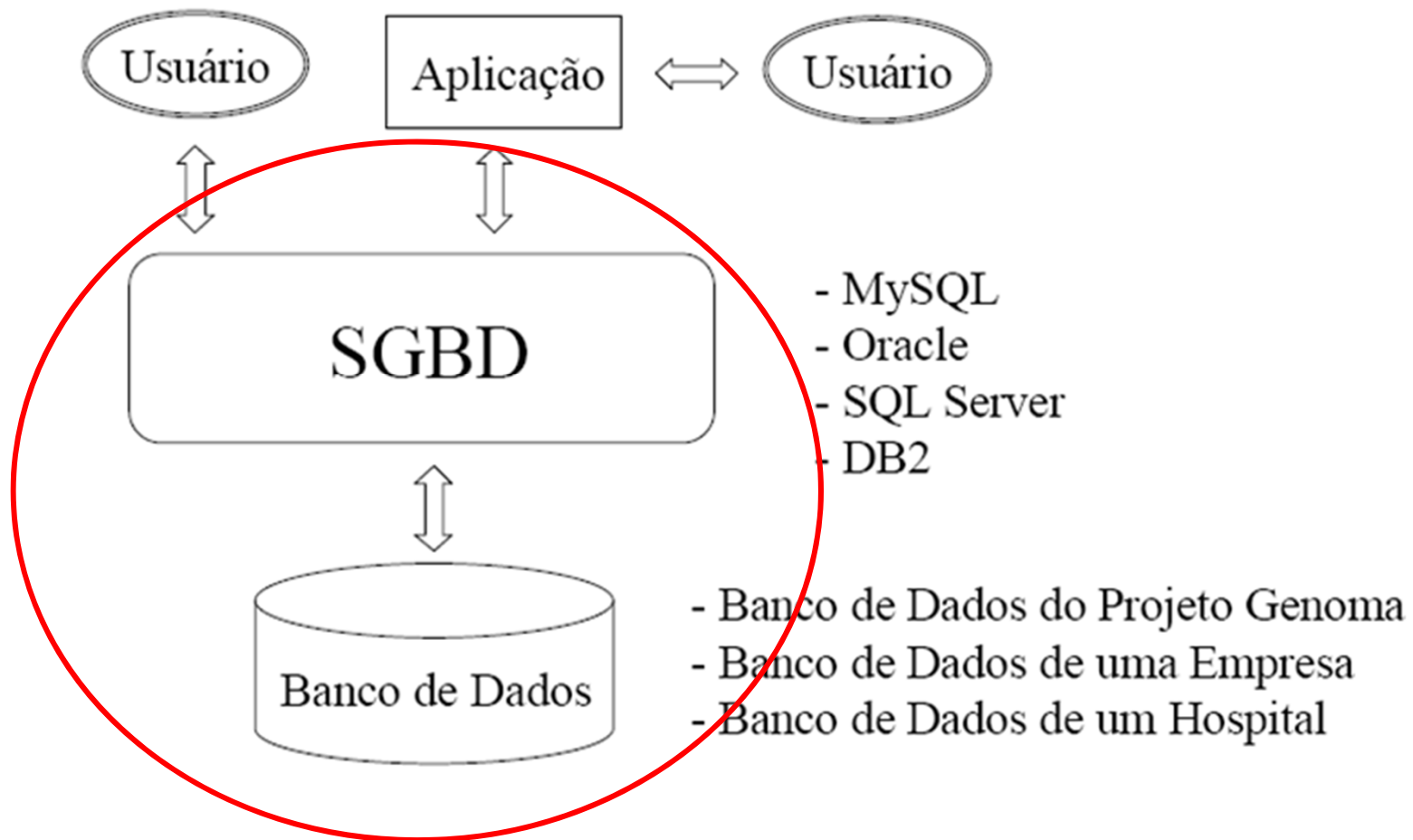
Disciplina: Sistemas de Banco de Dados  
Unidade IV: Projeto de Banco de Dados  
Prof.: Mario Filho  
E-mail: [pro@mariofilho.com.br](mailto:pro@mariofilho.com.br)  
Período: 5º. SIG - ADM

# Rápida Revisão do 1º Bimestre

# Conceitos

- **BD:** Coleção de dados inter-relacionados, projetada para atender às necessidades de um grupo específico de aplicações e usuários.
- **SGBD:** Sistema (conjunto de programas) de propósito geral que facilita o processo de construção e manipulação de bancos de dados para servir a diversas aplicações.

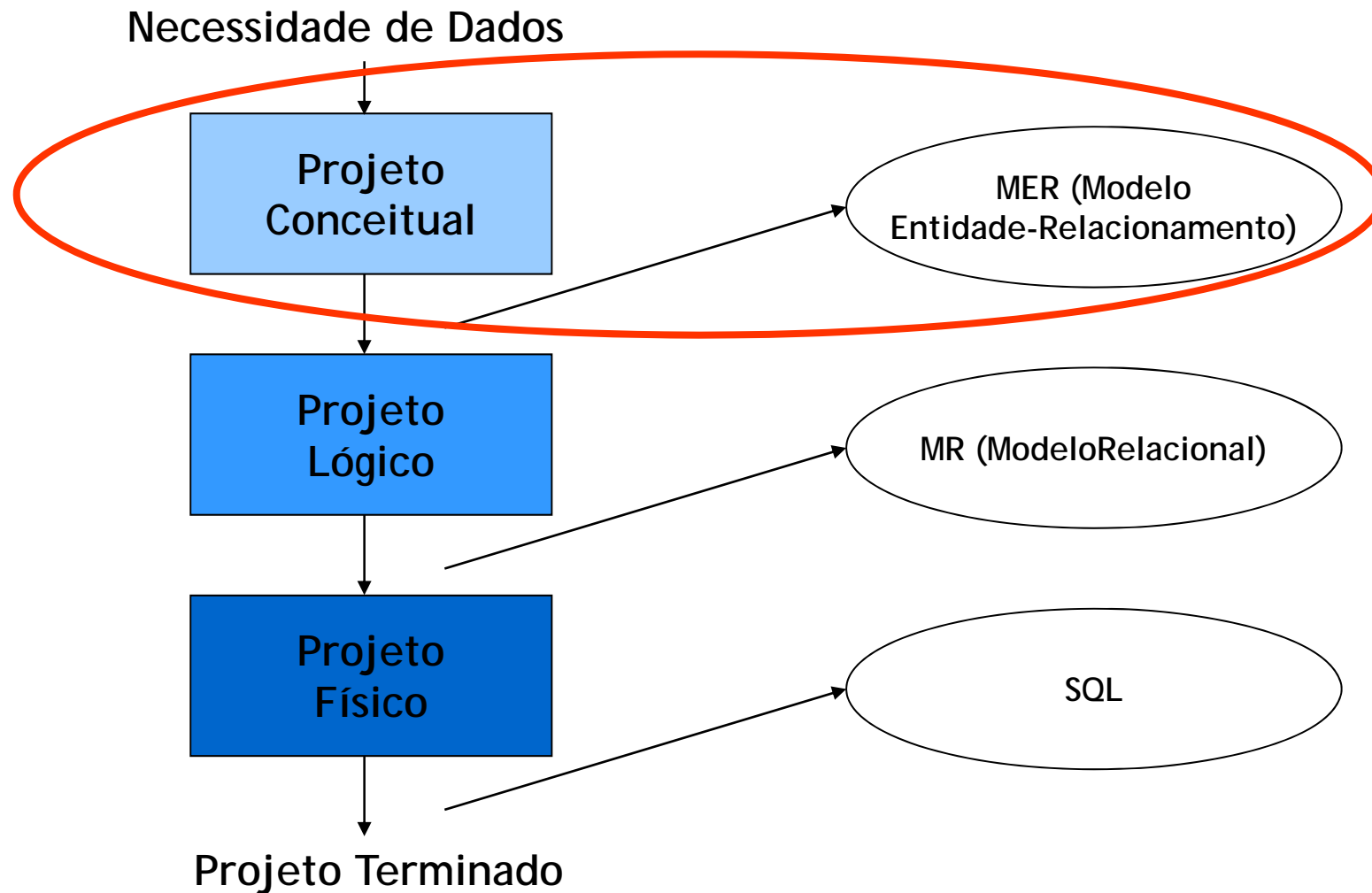
# Esquema



# Mais Conceitos

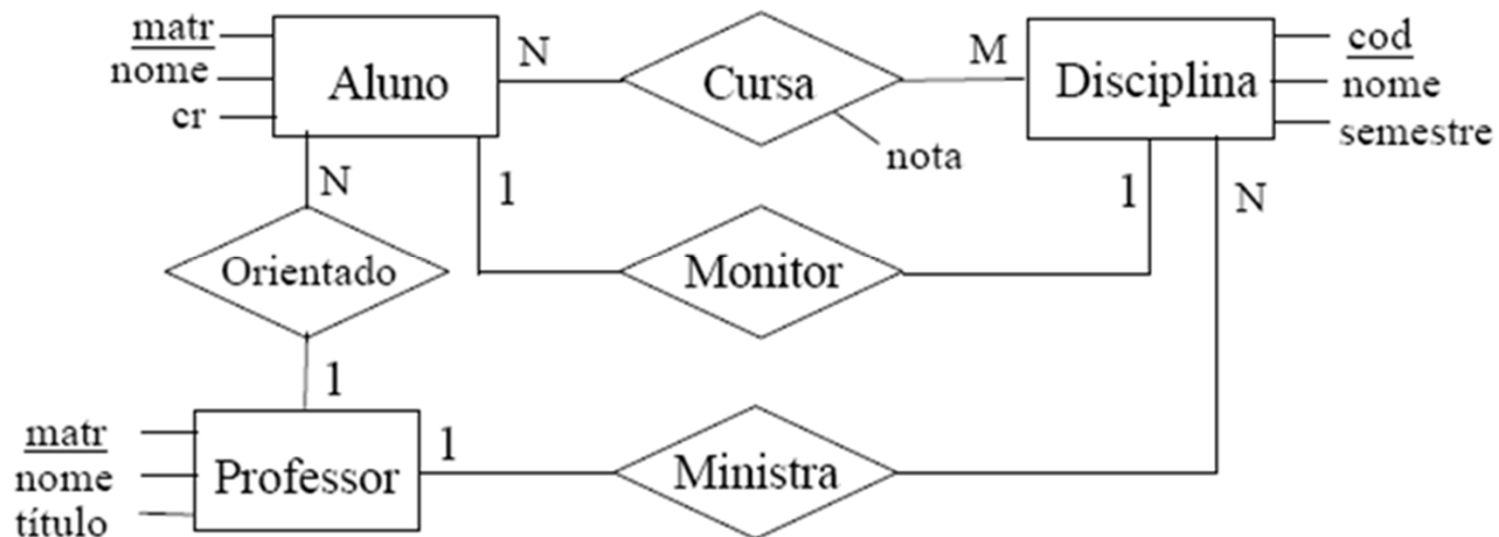
- Segurança em Banco de Dados:
  - Criptografia
  - Senhas
  - Backup
- Compartilhamento de Dados:
  - Redundância de Dados

# Etapas do Projetos de BD



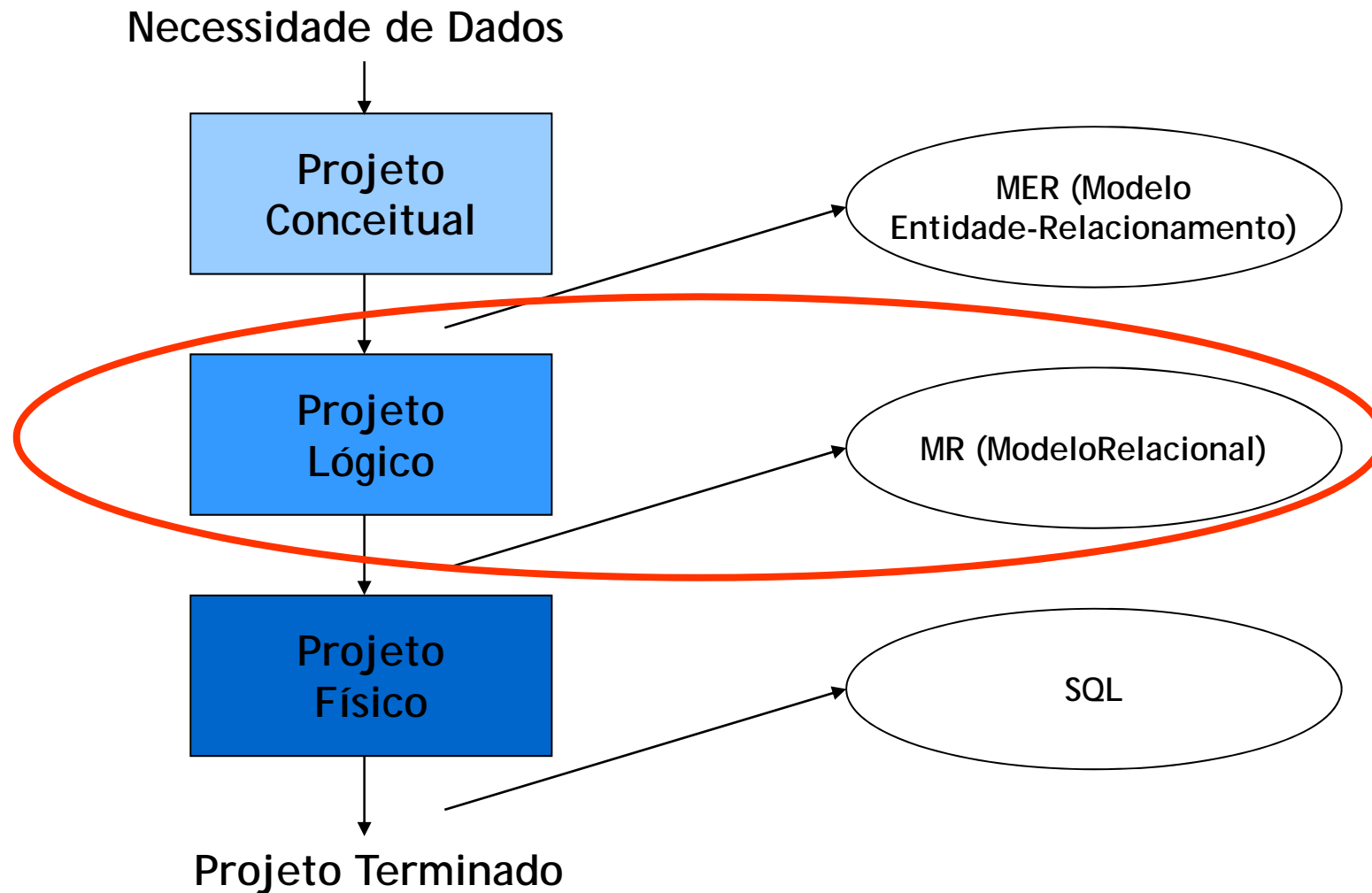
# Projeto Conceitual

- MER: Modelo Entidade-Relacionamento



*Conceitos, Notações Gráficas, PowerPoint e Trabalhos*

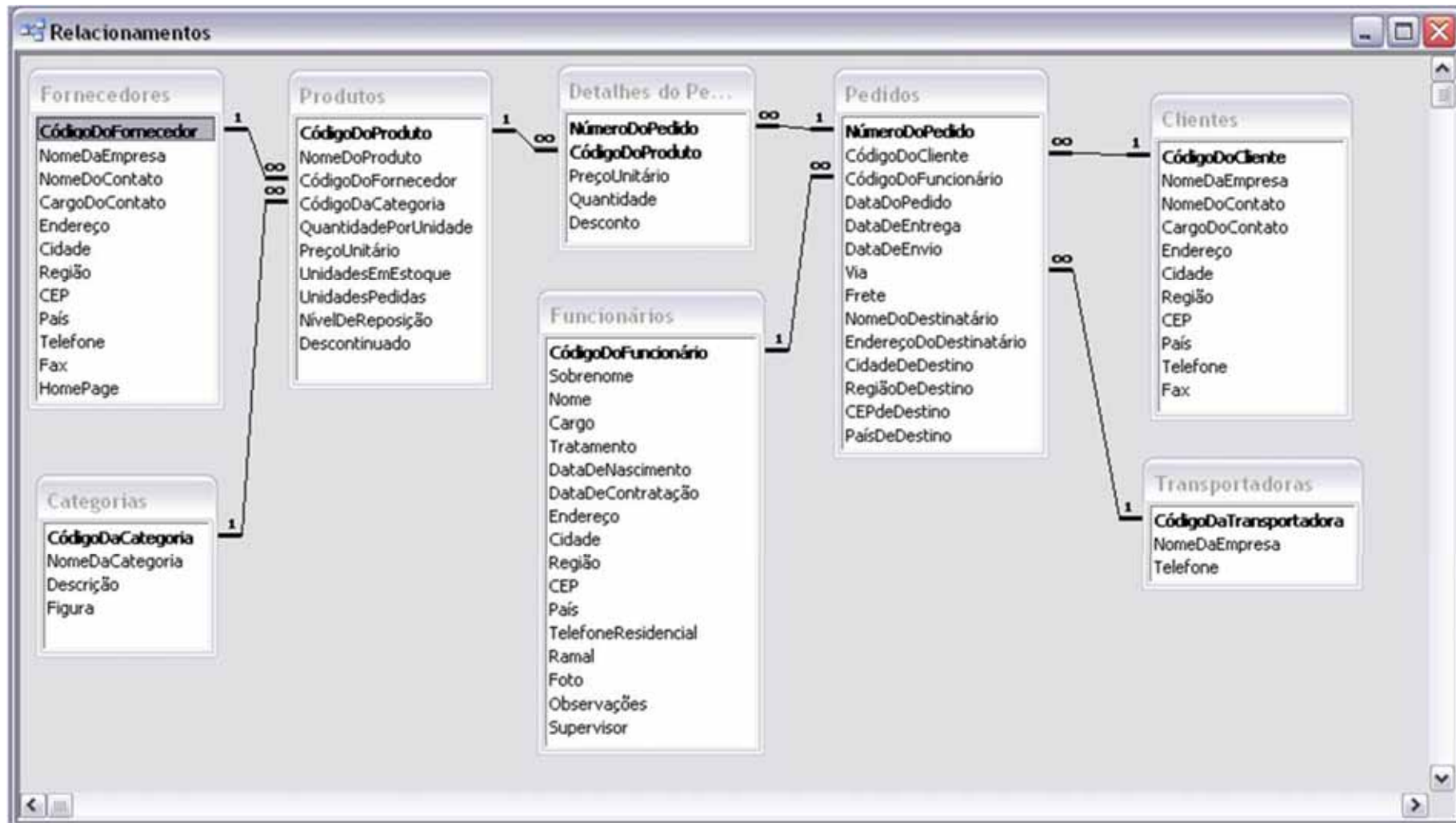
# Etapas do Projetos de BD



# Projeto Lógico

- MR ou Modelo Relacional
  - Entidade > Tabela
  - Atributo > Campo
  - Indexador > Chave Primária
  - Informações > Registros
  - Relacionamentos:
    - Um Para Um
    - Um Para Vários
    - Vários Para Vários
  - Integridade Referencial:
    - *"Propagar atualização dos campos relacionados"*.
    - *"Propagar exclusão dos registros relacionados"*.

# Modelo Relacional ou MR



Fim do 1º Bimestre

# 10. Dependências Funcionais



## 10.1. Definição

- Dado dois conjuntos de atributos (campos) A e B de uma entidade (tabela), diz-se que:
  - B é funcionalmente dependente de A ou...
  - A determina B ou...
  - B depende de A,se cada valor de A **estiver associado um, e só um**, valor de B.

## 10.1. Definição

- Se A determina B então B não é dependente funcional de nenhum subconjunto de A.
- Uma dependência funcional é representada por:
  - **A** → **B** (A: denominado Determinante)

# Exemplos

<u>CodFunc</u>	NomeFunc	Apelido	Departamento
1021	Milena	Lena	900
1022	Afonso	Reis	700
1023	Antonio	Tonho	900

- Departamento → CodFunc ?
  - NÃO, pois no Departamento 900: 1021,1023.
- CodFunc → Departamento ?
  - SIM, pois se conheço o CodFunc é possível determinar o Departamento em que ele trabalha.

# Mais Exemplos

<u>CodFunc</u>	NomeFunc	Apelido	Departamento
1021	Milena	Lena	900
1022	Afonso	Reis	700
1023	Antonio	Reis	900

- Apelido → CodFunc ?
  - NÃO, pois podem existir funcionários com o mesmo nome.
- CodFunc → Apelido ?
  - SIM, apesar de existirem dois funcionários com o mesmo apelido, se soubermos o CodFunc saberemos exatamente quem é o funcionário.
- CodFunc → todos os atributos restantes!

# 11. Normalização

# 11. Normalização de Tabelas

- **Objetivo:** evitar os problemas provocados por falhas no Projeto do Banco de Dados, bem como eliminar a **"mistura de assuntos"** e as correspondentes repetições desnecessárias de dados. Uma Regra de Ouro que devemos observar é a de **"não Misturar assuntos em uma mesma Tabela"**.
  - Exemplo: na Tabela Clientes devemos colocar somente campos relacionados com o assunto Clientes. Não devemos misturar campos relacionados com outros assuntos, tais como Pedidos, Produtos, etc. Essa "Mistura de Assuntos" em uma mesma tabela acaba por gerar repetição desnecessária dos dados bem como inconsistência dos dados.

# 11.1. Formas de Normalização

- O Processo de Normalização aplica uma série de Regras sobre as Tabelas de um Banco de Dados, para verificar se estas estão corretamente projetadas.
- Embora existam 5 formas normais (ou regras de Normalização), na prática usamos um conjunto de 3 Formas Normais (**1ªFN**, **2ªFN** e **3ªFN**).

# 11.1. Formas de Normalização

- Normalmente após a aplicação das Regras de Normalização, algumas tabelas acabam sendo divididas em duas ou mais tabelas, o que no final gera um número maior de tabelas do que o originalmente existente.
- Este processo causa a **simplificação** dos atributos de uma tabela, colaborando significativamente para a estabilidade do modelo de dados, reduzindo-se consideravelmente as necessidades de manutenção.
- Vamos entender o Processo de Normalização na Prática, através de exemplos...

# 1ª.FN (Primeira Forma Normal)

# Primeira Forma Normal (1ªFN)

- **Regra:** "Uma Tabela está na Primeira Forma Normal quando seus campos não contém grupos de Repetição".
  - Por isso, dizemos que uma Tabela que possui **Grupos de Repetição** NÃO está na Primeira Forma Normal (1ªFN).

# Primeira Forma Normal (1ªFN)

- Exemplo:



NomeDoAluno
NúmeroDaMatrícula
DataNascimento
Série
NomeDoPai
NomeDaMãe

Uma tabela com esta estrutura apresentaria diversos problemas. Por exemplo se um casal tiver mais de um filho, teremos que digitar o Nome do Pai e da Mãe diversas vezes, tantas quantos forem os filhos. Isso forma um Grupo de Repetição. Além do mais pode ser que por erro de digitação o Nome dos Pais não seja digitado exatamente igual todas as vezes, o que pode acarretar problemas na hora de fazer pesquisas ou emitir relatórios.

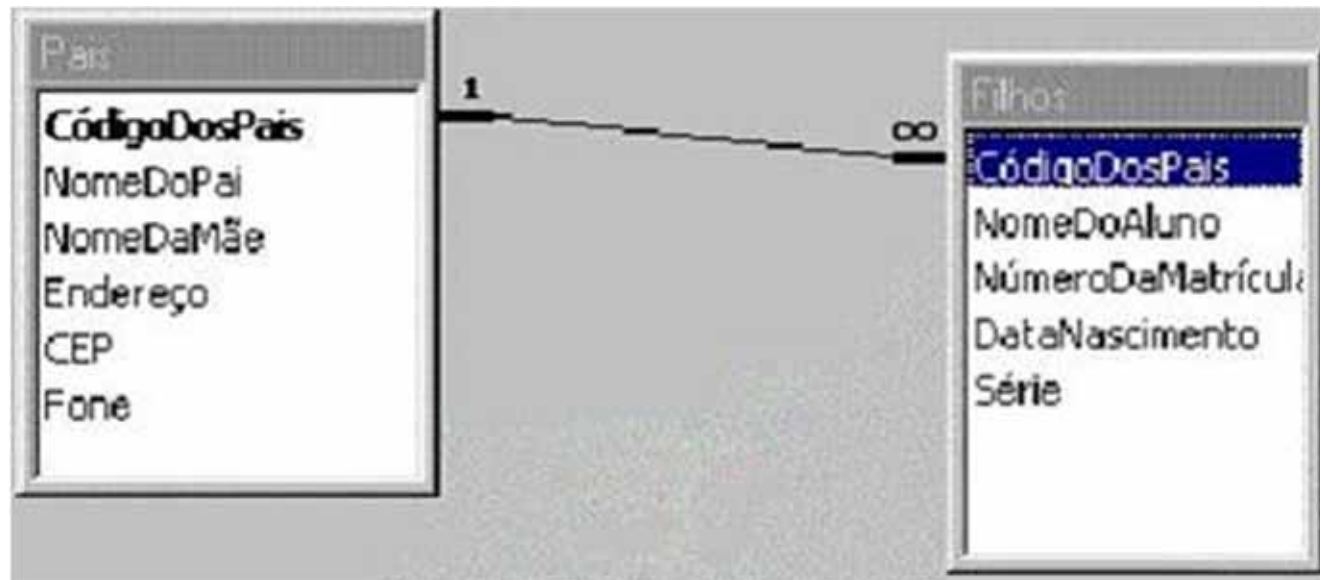
Conclusão: Esta Tabela NÃO está na 1ª.FN

# Primeira Forma Normal (1ªFN)

- Este problema ocorre porque "**Misturamos Assuntos**" em uma mesma tabela. Colocamos as informações dos Pais e dos Filhos em uma mesma tabela. A solução para este problema é simples...
  - Criamos uma tabela separada para a informação dos Pais e Relacionamos a tabela Pais com a Tabela Filhos através de um relacionamento do tipo Um para Vários, ou seja, um casal de Pais pode ter Vários Filhos.

# Primeira Forma Normal (1ªFN)

- **Solução:**



- As duas tabelas Resultantes da Aplicação da Primeira Forma Normal: **Pais** e **Filhos** agora **ESTÃO** na Primeira Forma Normal.
- A tabela original, a qual misturava informações de Pais e Filhos, **NÃO ESTAVA** na Primeira forma Normal

## 2<sup>a</sup>. FN (Segunda Forma Normal)

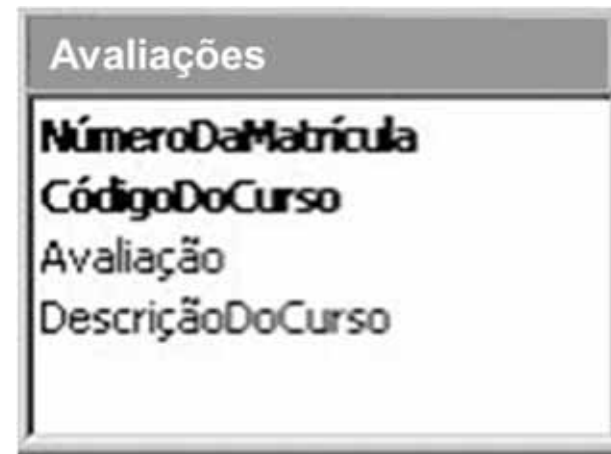
# Segunda Forma Normal (2ªFN)

- Só pode ser aplicada quando a Chave Primária é composta por mais de um campo.
- Neste caso, devemos observar se todos os campos que não fazem parte da chave dependem de todos os campos que compõem a chave. Se algum campo depender somente de parte da chave composta, então este campo deve pertencer a outra tabela.

Avaliações
<b>NúmeroDaMatricula</b>
<b>CódigoDoCurso</b>
Avaliação
DescriçãoDoCurso

# Segunda Forma Normal (2ªFN)

- A Chave Primária Composta é formada pela combinação dos Campos "NúmeroDaMatrícula" e "CódigoDoCurso".
- O Campo Avaliação depende tanto do CódigoDoCurso quanto do NúmeroDaMatrícula, porém o campo DescriçãoDoCurso, depende apenas do CódigoDoCurso, ou seja, dado o código do curso é possível localizar a respectiva descrição, independentemente do NúmeroDaMatrícula.
- Com isso temos um campo que não faz parte da Chave Primária e depende apenas de um dos campos que compõem a chave Primária Composta, por isso que dizemos que esta tabela **NÃO** está na Segunda Forma Normal.



# Segunda Forma Normal (2ªFN)

- **Solução:** "Dividimos a Tabela que não está na Segunda Forma Normal em duas outras tabelas, conforme indicado pela figura abaixo, sendo que as duas tabelas resultantes estão na Segunda Forma Normal.



- **Obs:** A Distinção entre a Segunda e a Terceira Forma Normal, que veremos logo em seguida, muitas vezes é confundida. **A Segunda Forma Normal está ligada a ocorrência de Chaves Primárias compostas.**

# 3<sup>a</sup>. FN (Terceira Forma Normal)

# Terceira Forma Normal (3ªFN)

- Na definição dos campos de uma tabela podem ocorrer casos em que um campo não seja dependente diretamente da chave primária ou de parte dela, mas sim dependente de um outro campo da tabela, campo este que não a Chave Primária.
- Quando isto ocorre, dizemos que a tabela não está na Terceira Forma Normal.

Funcionários não n...

<b>NúmeroDaMatricula</b>
NomeFuncionário
CódigoDoCargo
DescriçãoDoCargo

O diagrama mostra uma tabela com o título 'Funcionários não n...'. A primeira linha contém o campo 'NúmeroDaMatricula' em negrito, indicando que é a chave primária. As linhas subsequentes listam os campos 'NomeFuncionário', 'CódigoDoCargo' e 'DescriçãoDoCargo'. Este exemplo ilustra uma dependência transitiva onde 'CódigoDoCargo' depende de 'NúmeroDaMatricula' através de 'NomeFuncionário', o que viola a Terceira Forma Normal.

Não está na 3ªFN

## Terceira Forma Normal (3ªFN)

- Observe que o Campo **DescriçãoDoCargo** depende apenas do Campo **CódigoDoCargo**, o qual não faz parte da Chave Primária. Por isso dizemos que esta tabela não está na terceira forma normal.

Funcionários não n...	
<b>NúmeroDaMatricula</b>	
NomeFuncionário	
CódigoDoCargo	
DescriçãoDoCargo	

# Terceira Forma Normal (3ªFN)

- A Solução deste problema também é simples. Novamente basta dividir a tabela em duas outras, conforme indicado pela figura a seguir. As duas tabelas resultantes estão na Terceira Forma Normal.




Agora as duas tabelas estão na 3ªFN

## 8.2. Conclusão

- Com isso podemos concluir que como resultado do Processo de Normalização, iremos obter um número maior de tabelas, porém **sem problemas de redundância e inconsistência dos dados!!!**

# Exemplo Prático

## Aplicando as Regras



# Aplicando a 1ªFN

ProjFunc						
<u>CodProj</u>	DescProj	<u>CodFunc</u>	NomeFunc	CargoFunc	SalFunc	Data_Inicio
11	Alfa	1001	Antonio	Analista	5.000	01/02/08
		1004	João	Gerente	8.000	02/03/07
12	Beta	1003	Pedro	Analista	5.000	15/06/06

*1ªFN: Quando os campos da tabela NÃO contém grupos de repetição.*

*Solução 1ªFN:*

ProjFunc					
<u>CodFunc</u>	<u>CodProj</u>	NomeFunc	CargoFunc	SalFunc	Data_Inicio
1001	11	Antonio	Analista	5.000	01/02/08
1004	11	João	Gerente	8.000	02/03/07
1003	12	Pedro	Analista	5.000	15/06/06

PROJETOS	
<u>CodProj</u>	DescProj
11	Alfa
12	Beta

# Aplicando a 2ªFN

ProjFunc					
<u>CodFunc</u>	<u>CodProj</u>	NomeFunc	CargoFunc	SalFunc	Data_Inicio
1001	11	Antonio	Analista	5.000	01/02/08
1004	11	João	Gerente	8.000	02/03/07
1003	12	Pedro	Analista	5.000	15/06/06

*2ªFN : Chave Primária composta por mais de um campo? Se sim...  
Os campos que NÃO fazem parte da chave DEVEM DEPENDER dos que fazem parte*

Funcionários			
<u>CodFunc</u>	NomeFunc	CargoFunc	SalFunc
1001	Antonio	Analista	5.000
1004	João	Gerente	8.000
1003	Pedro	Analista	5.000

ProjFunc		
<u>CodFunc</u>	<u>CodProj</u>	Data_Inicio
1001	11	01/02/08
1004	11	02/03/07
1003	12	15/06/06

# Aplicando a 3ªFN

Funcionários			
<u>CodFunc</u>	NomeFunc	CargoFunc	SalFunc
1001	Antonio	Analista	5.000
1004	João	Gerente	8.000
1003	Pedro	Analista	5.000

*3ªFN : Todos os campos que não são chave, devem ser dependentes de um que seja. Nunca um campo que não é chave pode depender de um outro campo.*

Funcionários		
<u>CodFunc</u>	NomeFunc	CargoFunc
1001	Antonio	Analista
1004	João	Gerente
1003	Pedro	Analista

CargoSalário	
<u>CargoFunc</u>	SalFunc
Analista	5.000
Gerente	8.000

# Resultado Final

ProjFunc		
<u>CodFunc</u>	<u>CodProj</u>	Data_Inicio
1001	11	01/02/08
1004	11	02/03/07
1003	12	15/06/06

PROJETOS	
<u>CodProj</u>	DescProj
11	Alfa
12	Beta

Funcionários		
<u>CodFunc</u>	NomeFunc	CargoFunc
1001	Antonio	Analista
1004	João	Gerente
1003	Pedro	Analista

CargoSalário	
<u>CargoFunc</u>	SalFunc
Analista	5.000
Gerente	8.000

# Próximo Encontro:

- Exercícios e Exemplos de Normalização
- Instruções para o 1º. Trabalho do 2º. Bimestre.



Até segunda-feira!