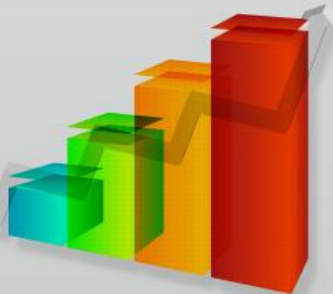


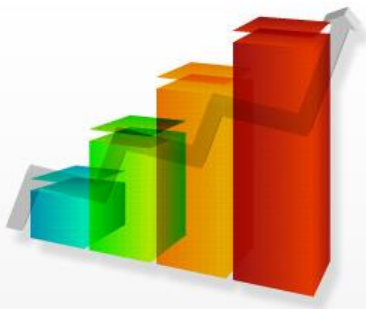
Introdução a Modelagem Lógica de Dados e Normalização de Banco de Dados

Gil Jader

Email: gil.jader@gmail.com

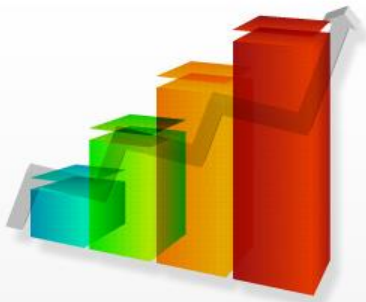
Salvador – Bahia
2013





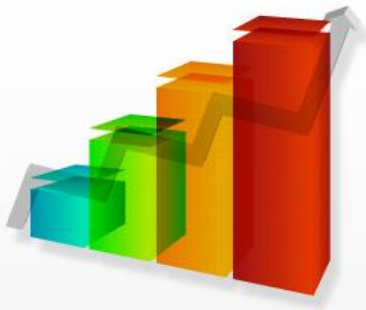
Indice

- Definições
- Gerações de Modelos Lógicos
- A Abordagem Relacional
 - Domínio
 - Tupla (Linha)
 - Relação
 - Tabela
 - Chave Candidata e Chave Primária (PK)
 - Chave Estrangeira (FK)
- Comparação da Nomenclatura (Modelo Conceitual X Modelo Lógico)
- Transformação Entre Modelos: Regras de Derivação
 - Derivação dos Relacionamentos
 - Relacionamento 1:1
 - Relacionamento 1:N
 - Relacionamento 1:N (Se houver atributo no relacionamento)
 - Relacionamento N:M
 - Relacionamento de Generalização/Especialização



Indice

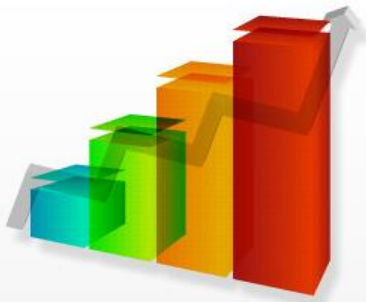
- Normalização de Banco de Dados e as Formas Normais
 - Definições
 - Anomalias de Banco de Dados
 - Primeira Forma Normal (1FN)
 - Dependência Funcional
 - Segunda Forma Normal (2FN)
 - Dependência Funcional Transitiva
 - Terceira Forma Normal (3FN)
 - Forma Normal de Boyce Codd
 - Quarta Forma Normal
 - Quinta Forma Normal



Modelo Lógico

- **Definição:**

- Modelo em que os objetos (classes de entidades), suas características (atributos) e relacionamentos têm representação de acordo com a regras de implementação e limitantes impostos por algum tipo de tecnologia;
- Esse modelo deve ser elaborado a partir de conceitos relacionados ao SGBD a ser usado se dá pela aplicação de regras de derivação sobre o modelo conceitual já construído.

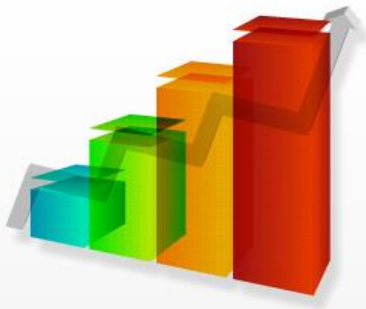


Modelo Lógico

- **Definição:**

Por que não partir diretamente para o modelo lógico ?

- Por que o modelo torna-se fortemente dependente do ambiente (tecnológico) onde será implantado. Isso acarretará a obtenção de modelos eficientes porém bastante dependentes da tecnologia que os orientou.
- Construindo um modelo conceitual de dados, pode-se derivá-lo para modelos lógicos para bancos de dados baseados em diferentes abordagens.

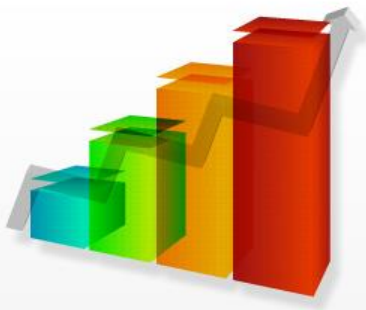


Modelo Lógico

▪ Gerações de Modelos Lógicos:

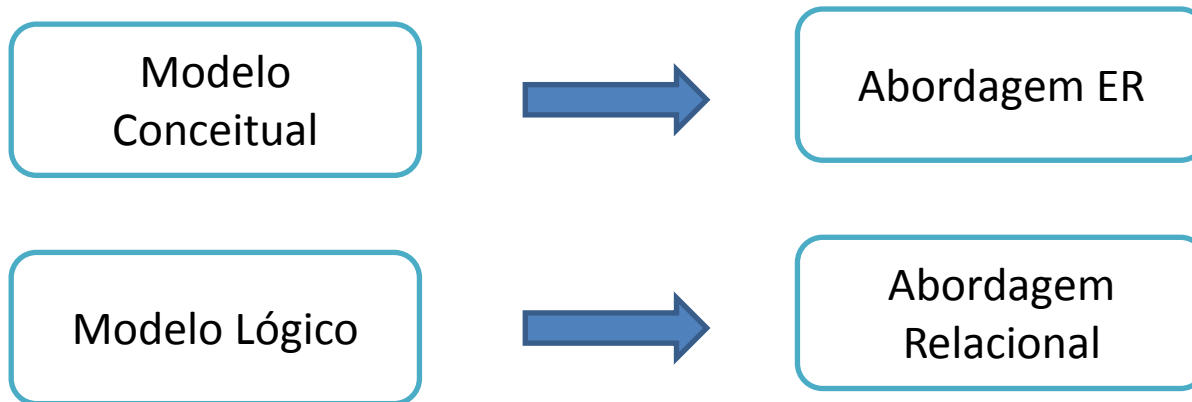
Linha do
tempo

- 1ª Geração: Modelos pré-relacionais
 - Modelo hierárquico e Modelo em rede
- 2ª Geração: Modelo relacional
- 3ª Geração: Modelos pós-relacionais
 - Modelo orientado a objetos, objeto-relacional, temporal, geográfico, etc.

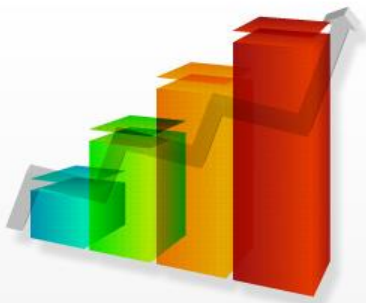


Modelo Lógico

- **Abordagem Relacional:**

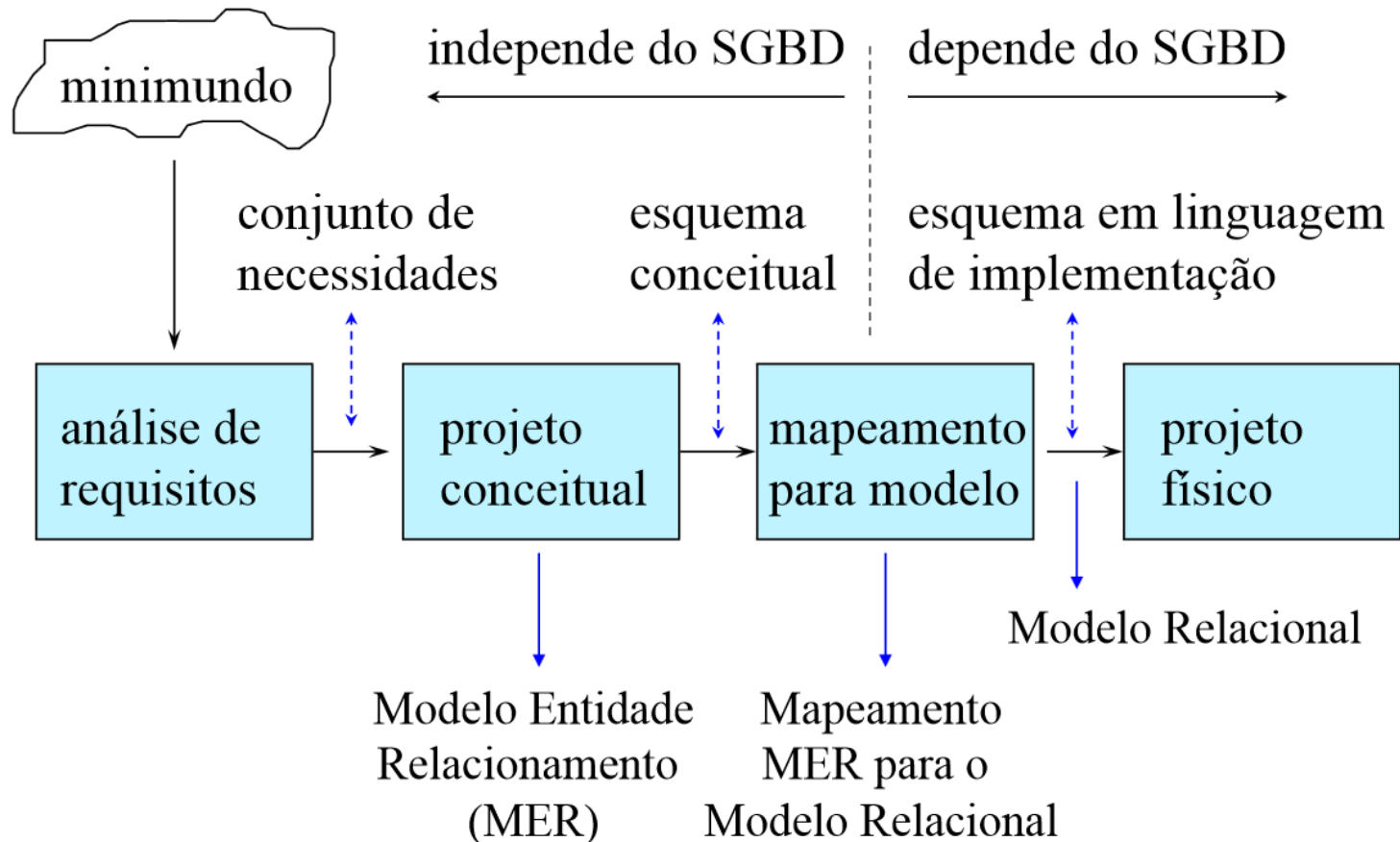


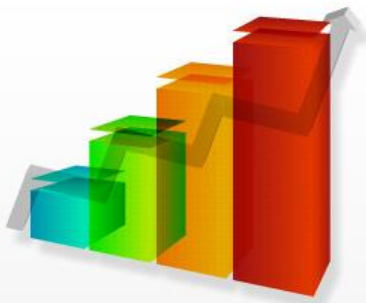
- Definida por Codd – década de 70.
- Modelo com sólida base formal: Teoria dos conjuntos



Modelo Lógico

▪ Modelo de Dados e o Projeto de BD

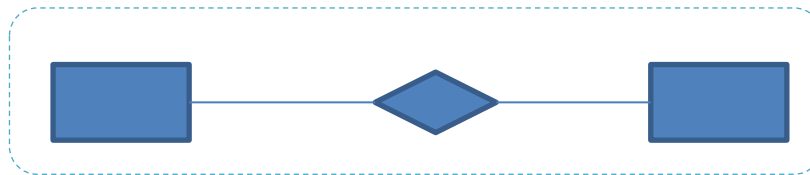




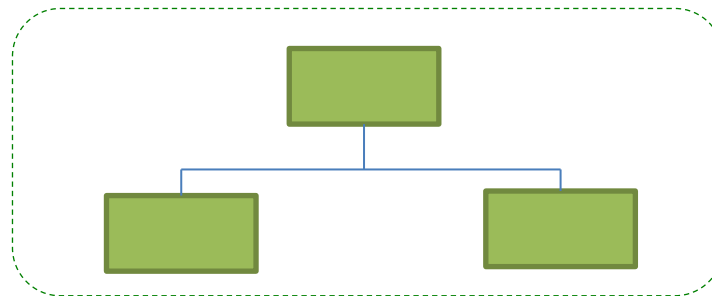
Modelo Lógico

▪ Representação dos Modelos / Etapas do Projeto de BD

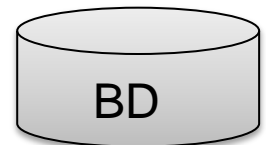
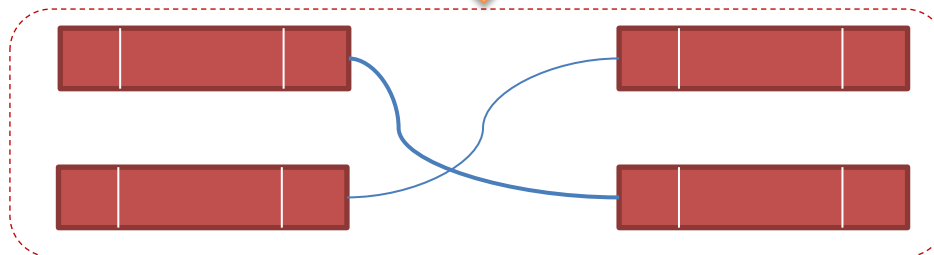
Modelo Conceitual
de Dados

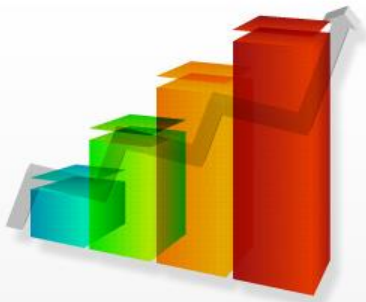


Modelo Lógico
de Dados



Modelo Físico
De Dados





Modelo Lógico

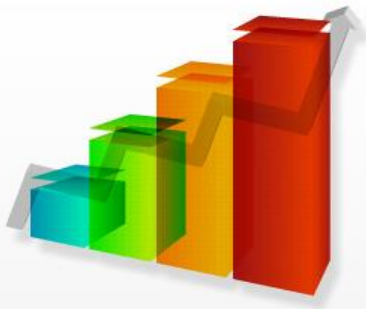
▪ Abordagem Relacional: principais conceitos

➤ Domínio

- Conjunto de valores permitidos para um campo
- Exemplos:
 - CPF: conjunto de 11 dígitos.
 - Idade: $16 \leq \text{idade} \leq 70$.

➤ Tupla/Linha/Registro

- Conjunto de pares (atributo, valor).
- Define uma ocorrência de um fato ou de um relacionamento entre fatos.
- Exemplo: Aluno: {(nome:'João'),(idade,31),(matrícula,031670034),...}



Modelo Lógico

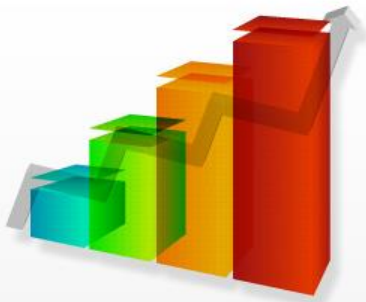
▪ Abordagem Relacional: principais conceitos

➤ Relação

- Elemento principal do modelo relacional, uma relação é uma tabela com linhas e colunas onde são armazenados os dados do banco de dados.

➤ Tabela

- Implementação das estruturas de dados.
- Representação plana de uma relação.



Modelo Lógico

- **Abordagem Relacional: principais conceitos**

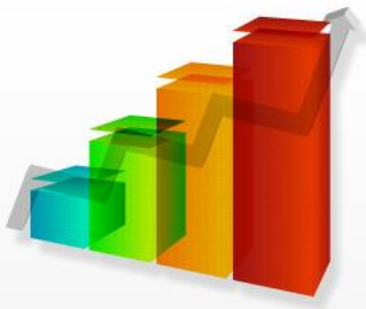
- Chave Candidata e Chave Primária (PK)

- Def. de relação: “... Não poderão existir 2 elementos iguais dentro de uma relação...”.

- Exemplo:

- **COLUNAS:** nome do eleitor, data de cadastramento, junta eleitoral, seção eleitoral, CPF, número do título de eleitor, data de nascimento, etc.
 - **CHAVES CANDIDATAS:** número título de eleitor, número CPF.
 - **CHAVES PRIMÁRIA (PK):** número CPF.

Por quê ? Natural do ambiente no qual será usado.



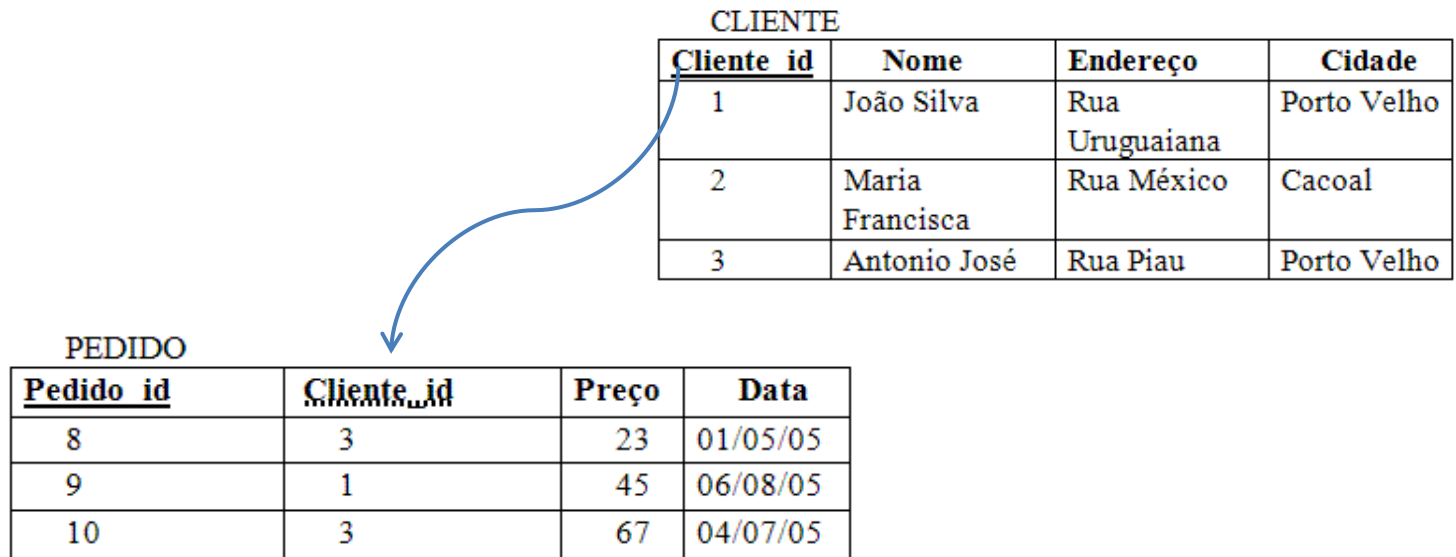
Modelo Lógico

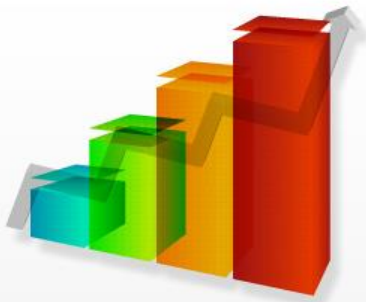
▪ Abordagem Relacional: principais conceitos

➤ Chave Estrangeira (FK)

- É o método de estabelecimento do relacionamento entre duas tabelas.

Exemplo:



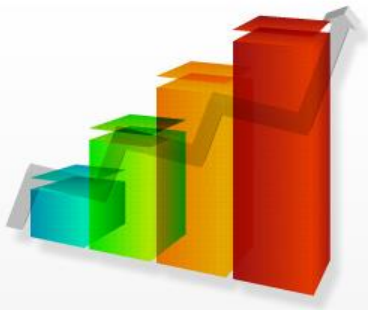


Modelo Lógico

- **Abordagem Relacional: principais conceitos**

- Nomenclatura (modelo conceitual X modelo lógico relacional)

Modelo Conceitual	Modelo Lógico Relacional
Entidade	Tabela
Atributo	Coluna/Campo (Domínio)
Relacionamento	Relacionamento (Relação)
Instância	Linha/Registro (Tupla)

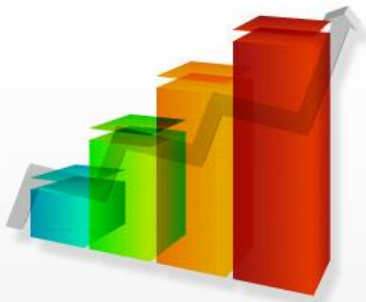


Modelo Lógico

- **Esquema Textual**

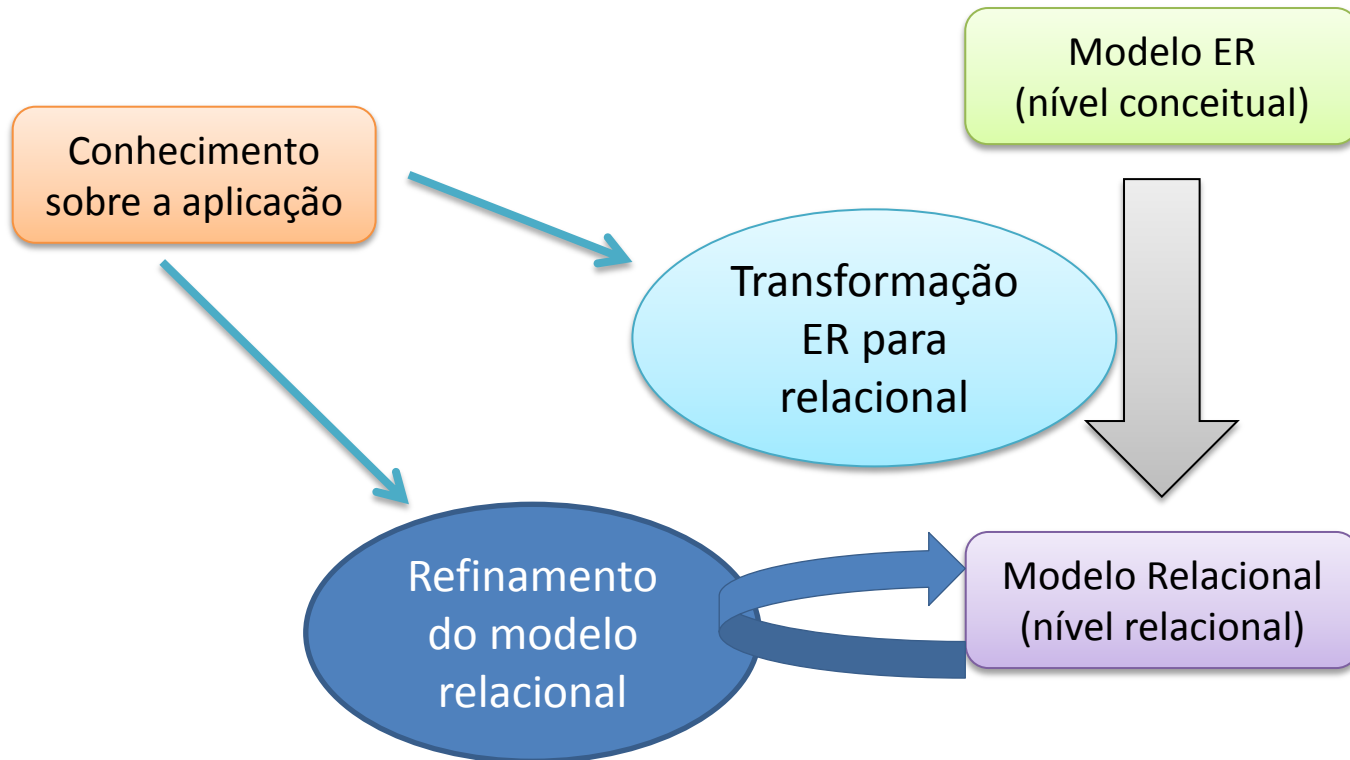
Emp (CodigoEmp, nome, CodigoDepto, CategFuncional, CPF)

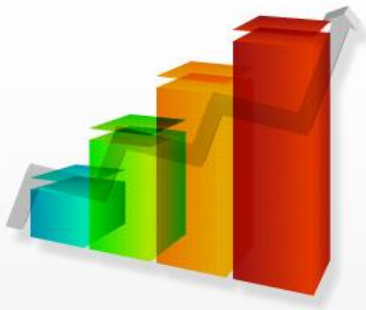
Dept (CodigoDepto, nome)



Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**





Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

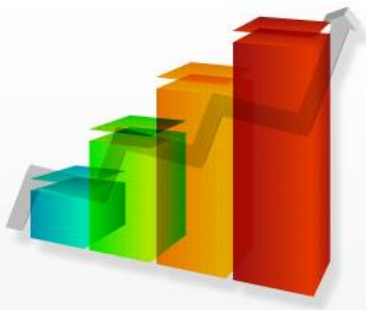
- Derivação dos Relacionamentos

- Relacionamento 1:1

A) **Adição de Coluna** (Acrescentar a chave da tabela A como chave estrangeira da tabela B ou vice versa)

OU

B) **Fusão de Tabelas** (Migrar os atributos e relacionamentos da tabela A para a tabela B, excluindo a tabela A ou vice versa)

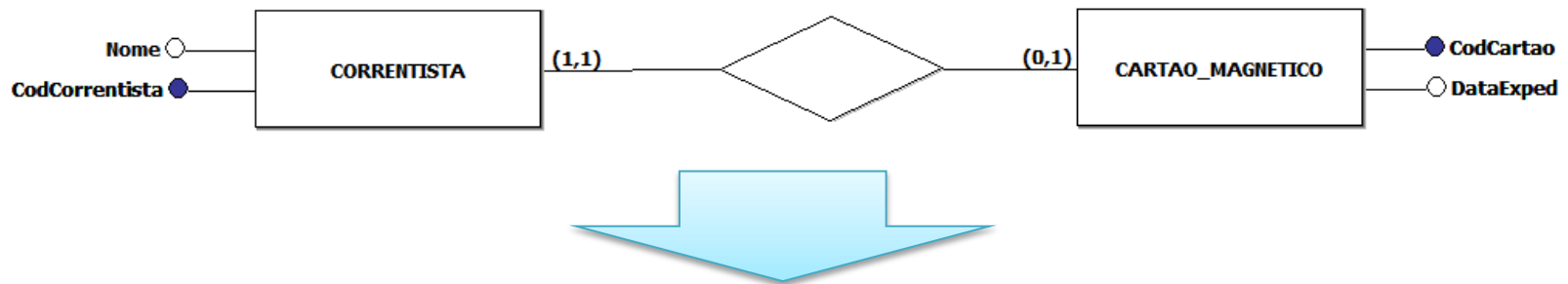


Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

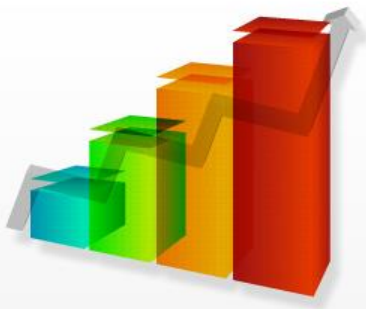
- Derivação dos Relacionamentos

- Relacionamento 1:1



A) **Adição de Coluna** (Acrescentar a chave da tabela A como chave estrangeira da tabela B ou vice versa)

Correntista(CodCorrentista, Nome)
Cartao(CodCartao, CodCorrentista, DataExped,)



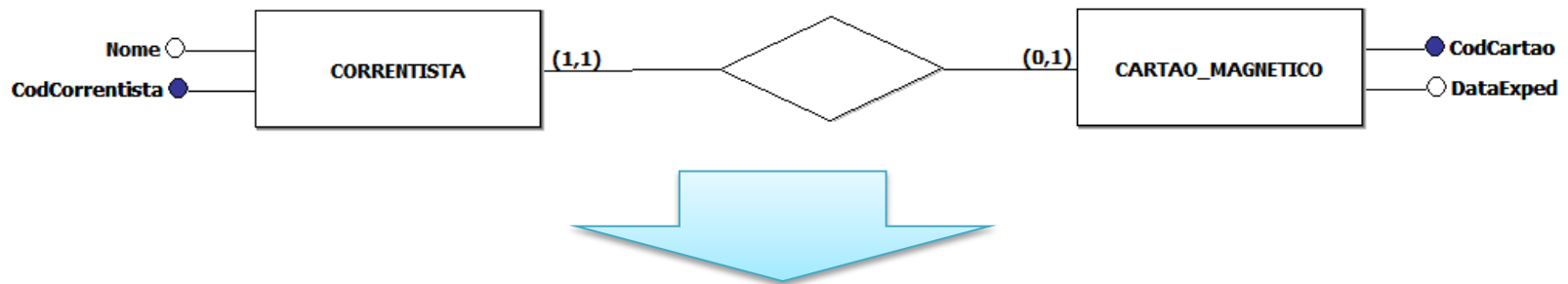
Modelo Lógico

▪ Transformações Entre Modelos: Regras de Derivação:

➤ Derivação dos Relacionamentos

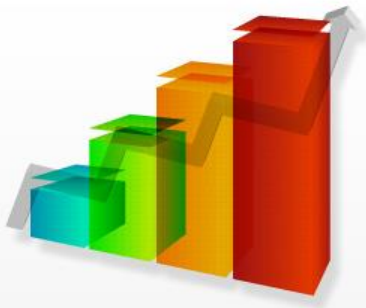
▪ Relacionamento 1:1

Segunda Opção



B) **Fusão de Tabelas** (Migrar os atributos e relacionamentos da tabela A para a tabela B, excluindo a tabela A ou vice versa)

Correntista(CodCorrentista, Nome, CodCartao, DataExped)



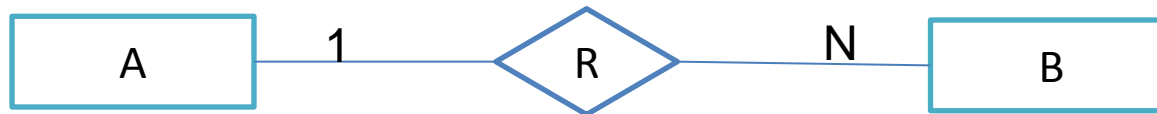
Modelo Lógico

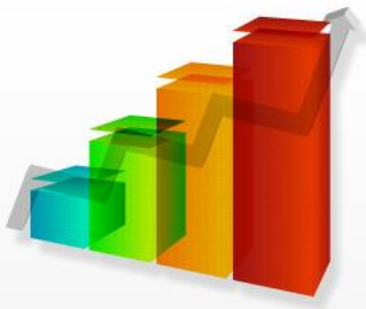
- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento 1:N

- Acrescentar a chave da tabela A (lado “1” do relacionamento) como chave estrangeira da tabela B (lado “N” do relacionamento)
- A simples inclusão da chave estrangeira da tabela A na tabela B faz com que o relacionamento de 1:N fique implementado satisfatoriamente



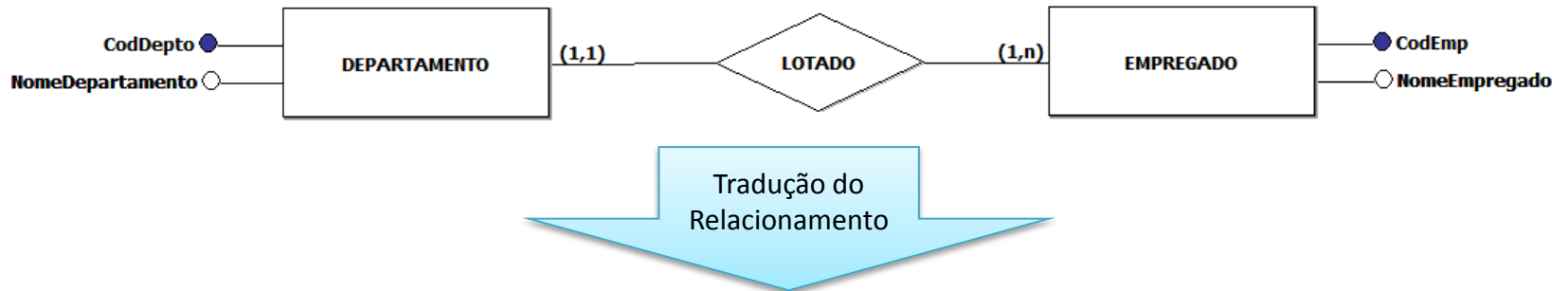


Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

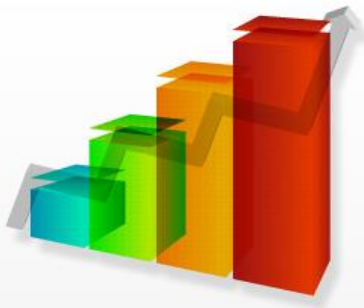
- Derivação dos Relacionamentos

- Relacionamento 1:N



Departamento(CodDepto, Nome)

Empregado (CodEmp, CodDepto, NomeEmpregado, NomeDepartamento)

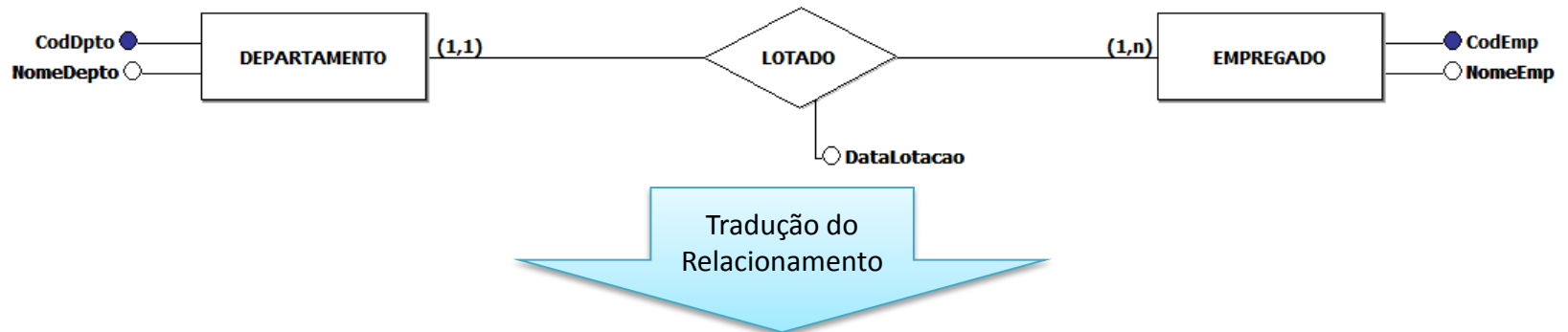


Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

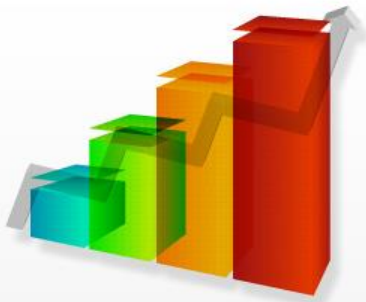
- Relacionamento 1:N – Se houver atributos no relacionamento



Departamento(CodDeppto, Nome)

Empregado (CodEmp, NomeEmp)

Lotacao (CodEmp, CodDeppto, DataLotacao)



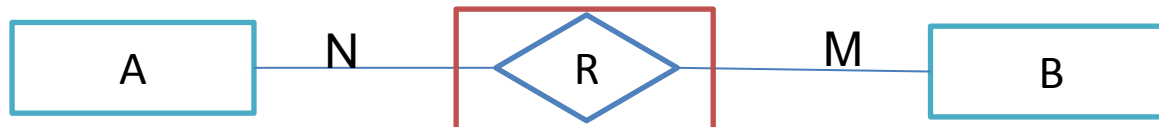
Modelo Lógico

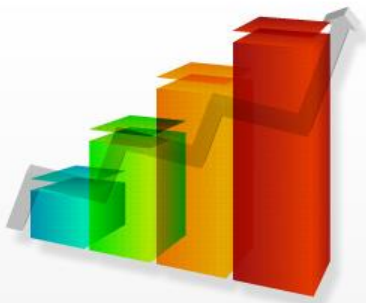
- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento N:M

- Cria-se a terceira tabela C (entidade associativa) agregando as chaves estrangeiras de A e B para formar a chave primária da tabela C.



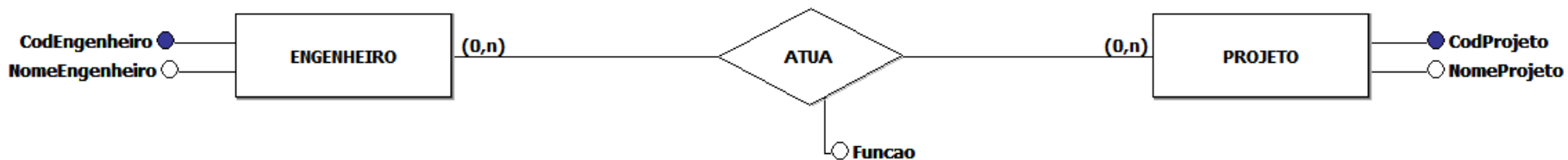


Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento N:M

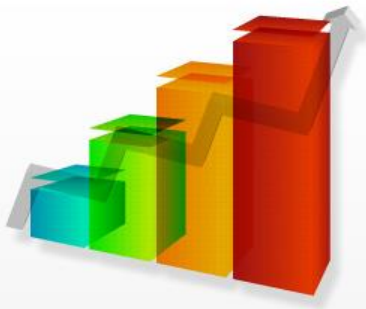


Tradução do
Relacionamento

Engenheiro(CodEngenheiro, NomeEngenheiro)

Projeto(CodProjeto, NomeProjeto)

Lotacao (CodEngenheiro, CodProjeto, Funcao)



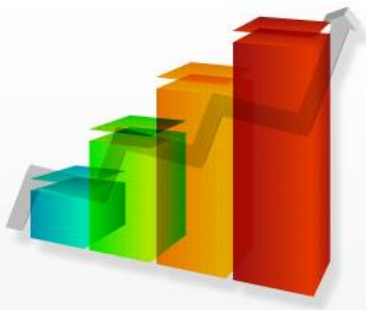
Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento de generalização/especialização

- a) Uso de uma tabela para cada entidade;
 - b) Uso de uma tabela para toda a hierarquia;



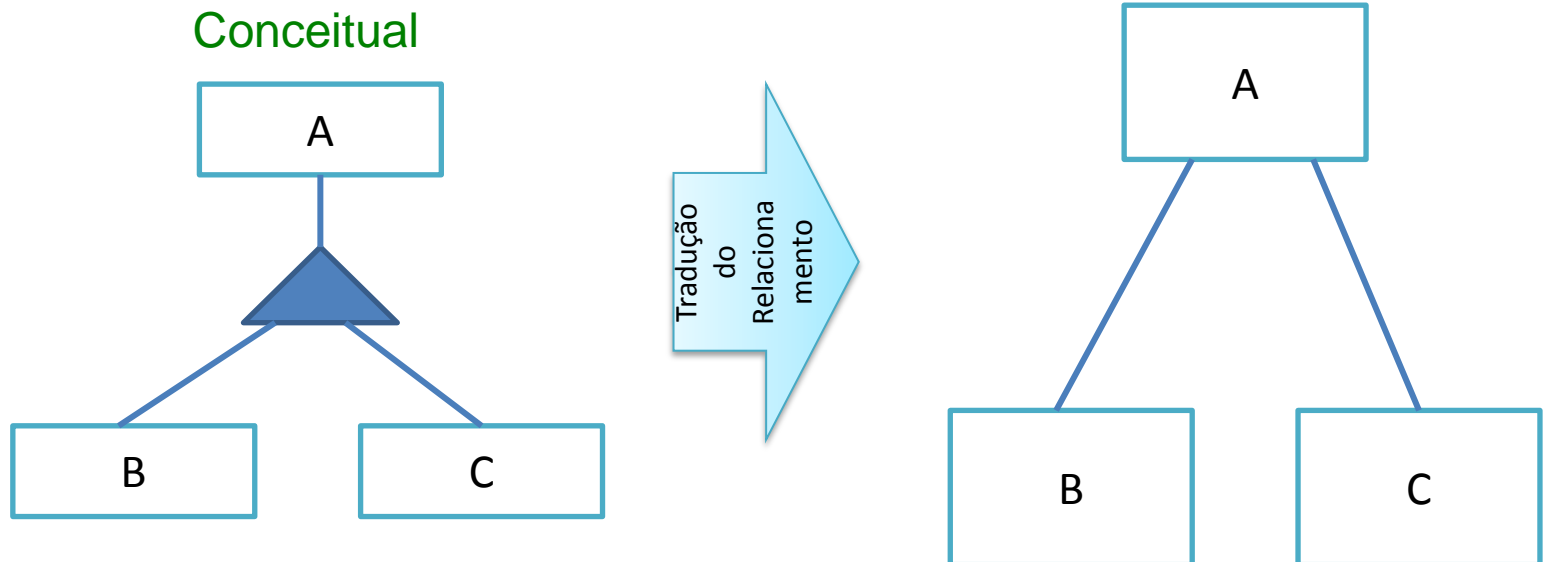
Modelo Lógico

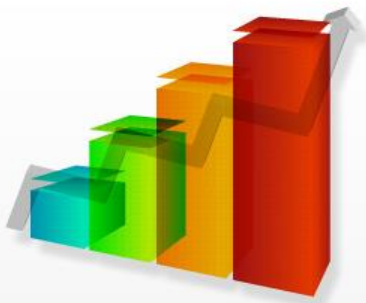
- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento de generalização/especialização

- a) Uso de uma tabela para cada entidade





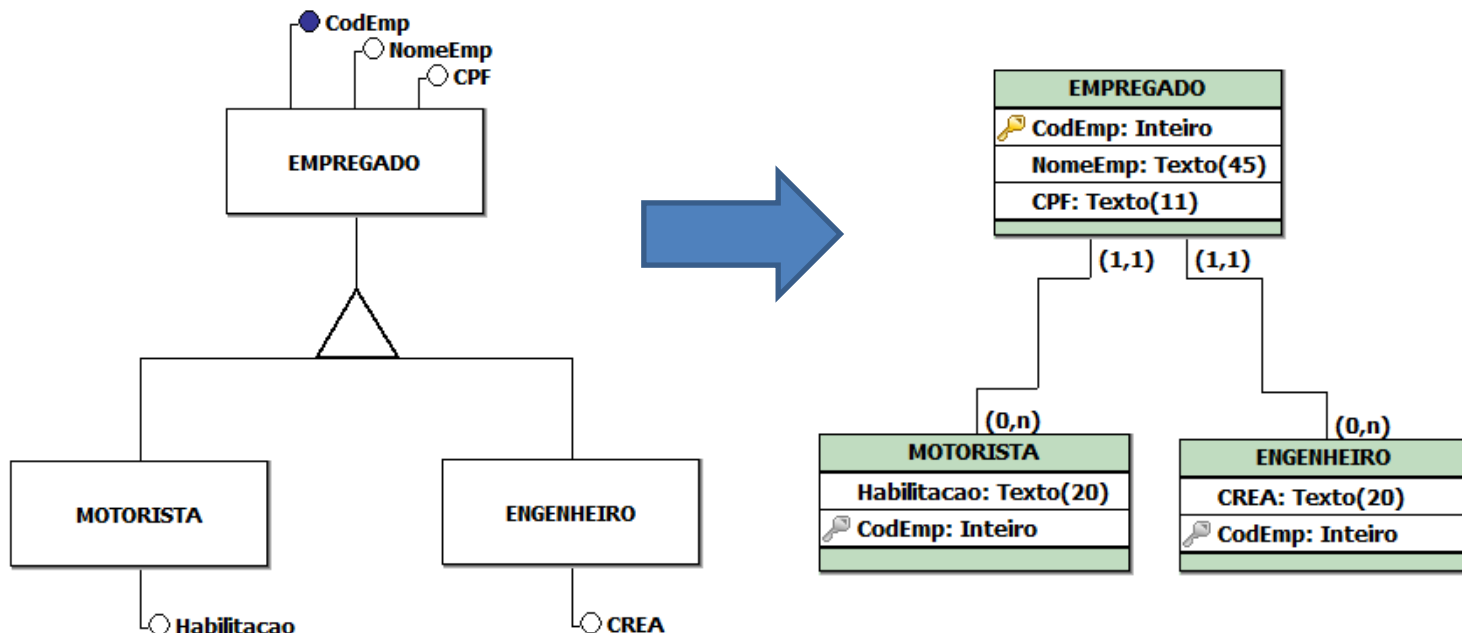
Modelo Lógico

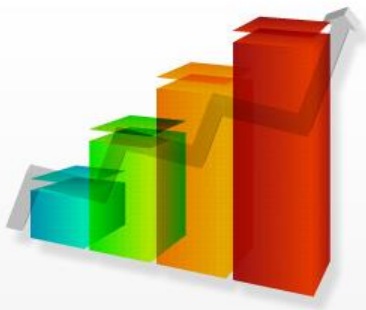
- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento de generalização/especialização

a) Uso de uma
tabela para cada
entidade





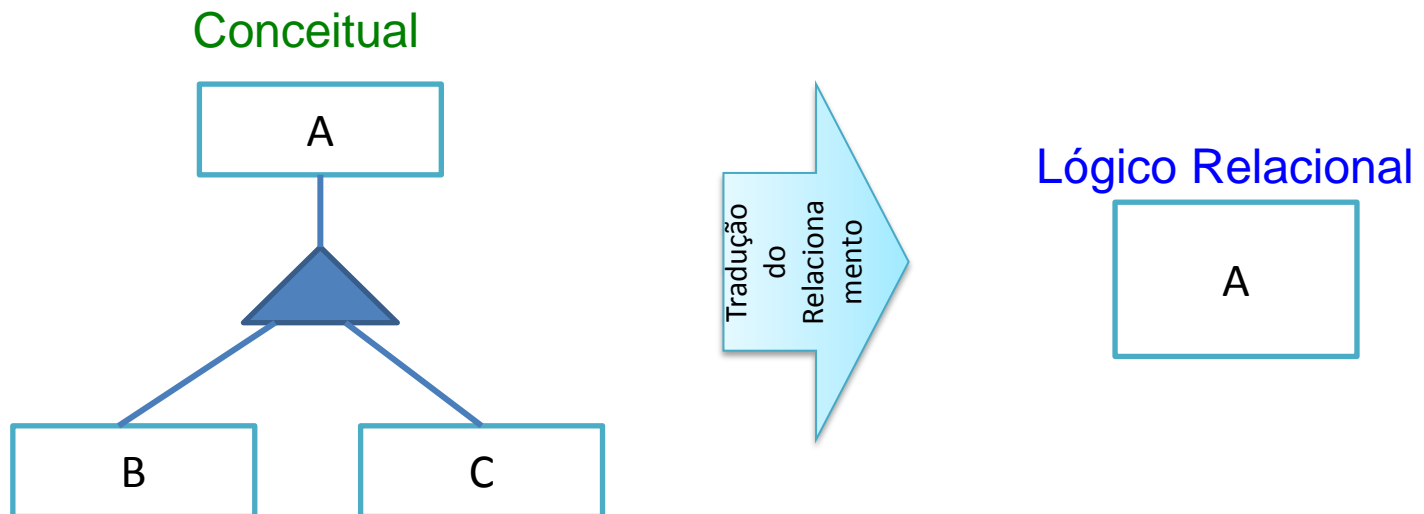
Modelo Lógico

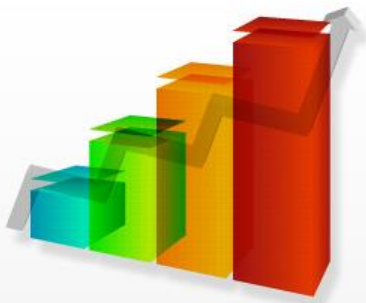
- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

- Relacionamento de generalização/especialização

- a) Uso de uma tabela para toda a hierarquia





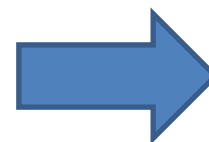
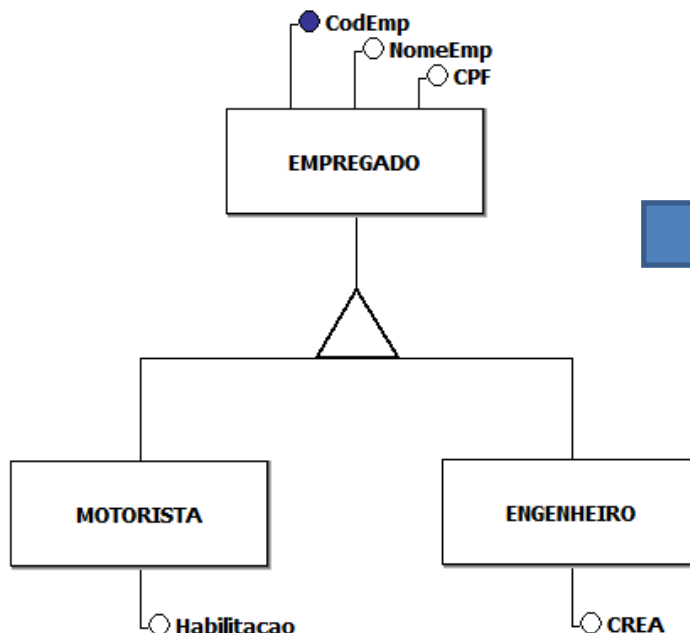
Modelo Lógico


- **Transformações Entre Modelos: Regras de Derivação:**

- Derivação dos Relacionamentos

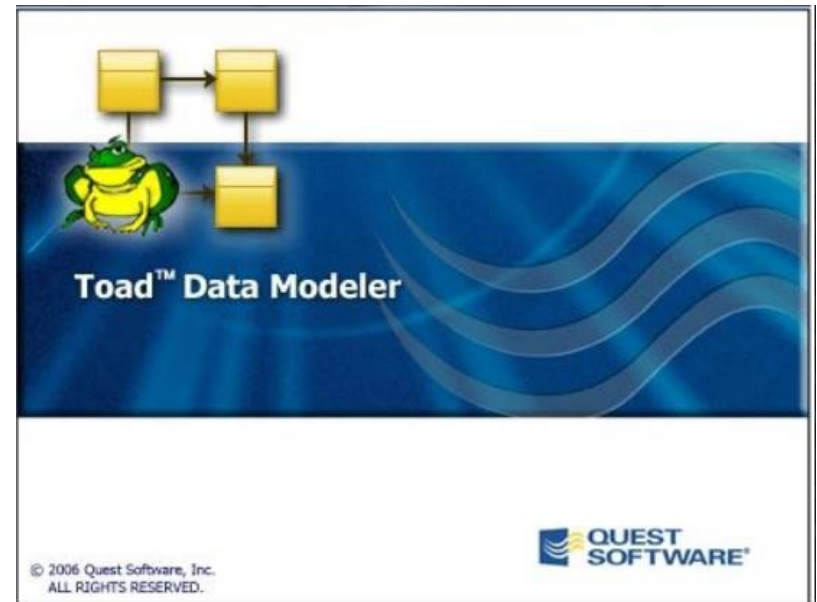
- Relacionamento de generalização/especialização

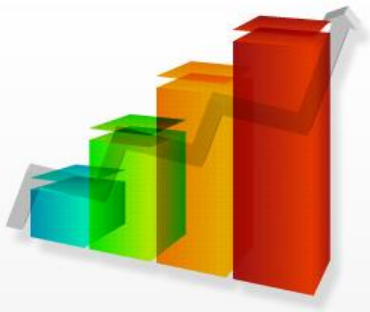
a) Uso de uma
tabela para cada
entidade



EMPREGADO+MOTORIST...	
	CodEmp: Inteiro
	NomeEmp: Texto(45)
	CPF: Texto(11)
	Habilitacao: Texto(20)
	CREA: Texto(20)

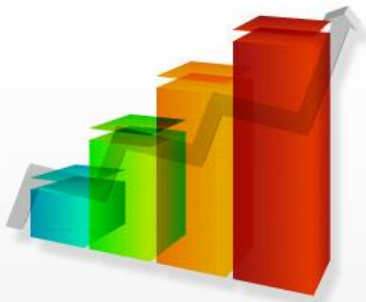
Exercícios Modelo Lógico de Dados





Modelo Lógico

Normalização de Dados



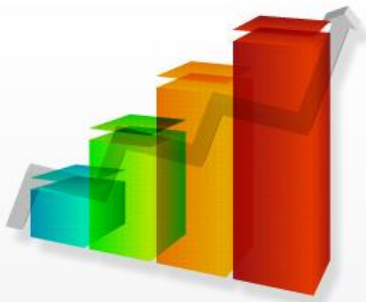
Modelo Lógico

- **Normalização de Dados**

- Definições

- Processo formal que examina campos de uma tabela com objetivo de racionalizar as estruturas de dados de um sistema;
- O objetivo da normalização é evitar os problemas que podem provocar falhas no projeto lógico do banco de dados, afim de evitar falhas no projeto, como **redundância de dados**, **mistura de diferentes assuntos numa mesma tabela**, entre outros problemas.

Não Misturar Assuntos em Uma Mesma Tabela



Modelo Lógico

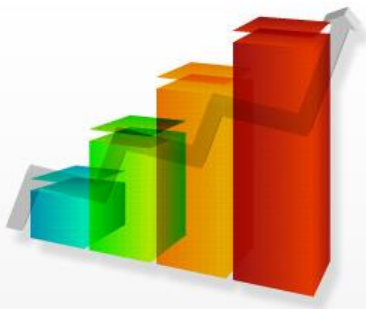
- **Normalização de Dados**

- Definições

- Diz-se que a tabela que atende a um determinado conjunto de regras está em uma determinada FORMA NORMAL (FN).

Portanto:

- A normalização consiste em definir o formato lógico adequado para as estruturas de dados das tabelas de um banco de dados relacional, identificadas no projeto lógico do sistema, com o objetivo de minimizar o espaço utilizado pelos dados e garantir a integridade e confiabilidade das informações.



Modelo Lógico

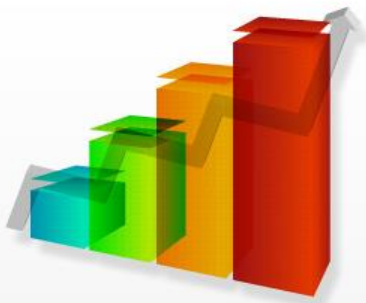
▪ Normalização de Dados

- Quais problemas podem ocorrer com a tabela de vendas abaixo ?

Nome	<u>CPF</u>	End	Tel	<u>Pcod</u>	Nprod	Vunit	Qtd	Tot
Zé	111	ABC	123	A	Lápis	0,50	2	1,00
Ana	222	XYZ	456	B	Caneta	1,00	3	3,00
João	333	XPT	789	C	Régua	1,00	2	2,00
Pedro	444	KZZ	NULL	A	Lápis	0,50	20	10,00

*Causa fatos conhecidos como **anomalias** de **inclusão**, **alteração** e **deleção** de dados:*

- Repete informações a cada nova inclusão;
- Perde completamente as informações do registro quando existe exclusão;
- Alteração tem que varrer todos os registros para encontrar os valores e alterar;



Modelo Lógico

- **Normalização de Dados**

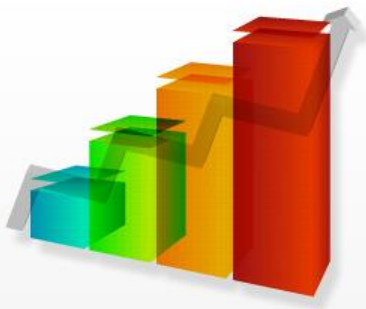
- Anomalias de Banco de Dados - Exemplos

Tabela Alunos Academia

Nome	Atividade	Taxa
José	Musculação	30,00
Pedro	Judô	35,00
Manoel	Karatê	45,00

1º Se excluirmos o aluno Manoel, o que acontece ? **Anomalia de Eliminação**

Excluimos não apenas o nome do aluno, mas também as informações referentes as atividades de musculação, bem como o seu valor.



Modelo Lógico

- **Normalização de Dados**

- Anomalias de Banco de Dados - Exemplos

Tabela Alunos Academia

Nome	Atividade	Taxa
José	Musculação	30,00
Pedro	Judô	35,00
Manoel	Karatê	45,00

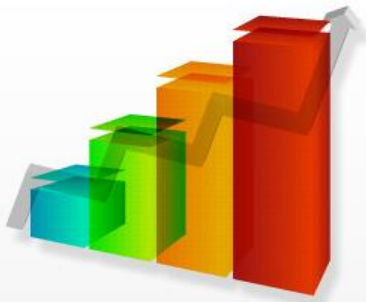
	jiu jitsu	100,00
--	-----------	--------



2º Quero incluir um novo curso na academia, o que acontece ?

Anomalia de Inserção

Não podemos inseri-lo até que um aluno tenha disponibilidade para fazê-lo.



Modelo Lógico

- **Normalização de Dados**

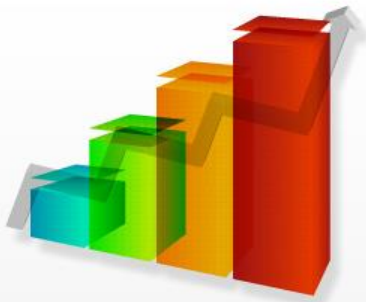
- Anomalias de Banco de Dados - Exemplos

Tabela Alunos Academia

Nome	Atividade	Taxa
José	Musculação	30,00
Pedro	Judô	35,00
Manoel	Judo	35,00

3º Se atualizarmos a atividade de Manoel para “*Judo*”, o que acontece ? **Anomalia de Atualização**

Note, que Judô, está grafado de forma errada na tupla do aluno Manoel. Se uma busca for feita por Judô, só irá aparecer 1 aluno e não 2 alunos.



Modelo Lógico

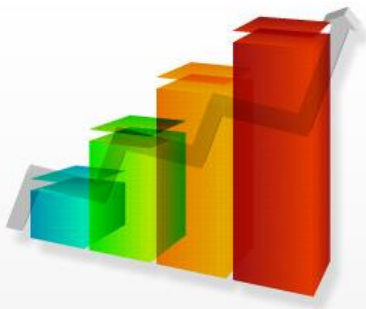
- **Normalização de Dados**

- PRIMEIRA FORMA NORMAL(1FN):

Uma Tabela está na **1FN** se e somente se todos os seus campos contem apenas valores atômicos (simples).

Ou seja:

- Existe uma chave primária;
- Não existem tuplas repetidas e nem campos multivalorados



Modelo Lógico

- **Normalização de Dados**

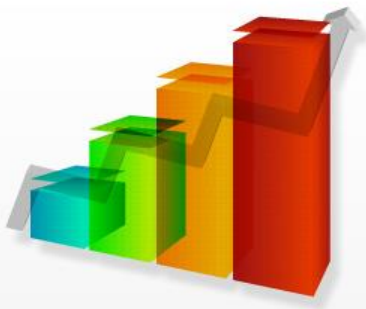
- PRIMEIRA FORMA NORMAL(1FN):

Primeiro exemplo:

idCliente	NomeCliente	Endereco	Telefones
1	Jose da Silva	Rua Justiano, 3300	2342-2342 / 2242-1210
2	Emanuel Bezerra	Travessa Quinze, 200	8952-2231
3	João dos Santos	Av. Brasil, 220	6758-5391 / 4302-9000 / 7548-9091
4	Maria Mariana	Av. Bandeirantes, 2908	9099-3022

➤ **Passos e maneira mais conhecida para colocar a Tabela na 1FN:**

- 1- Identificação da chave primária da tabela;
- 2- Identificação da(s) coluna(s) que contem dados multivalorados e removê-las. Excluir também as tuplas com informações duplicadas, deixando apenas uma;
- 3- Criação de uma nova tabela com chave primária para armazenamento do dado repetido;
- 4- Criar uma relação entre a tabela principal e a secundária;



Modelo Lógico

- **Normalização de Dados**

- PRIMEIRA FORMA NORMAL (1FN):

Primeiro exemplo:

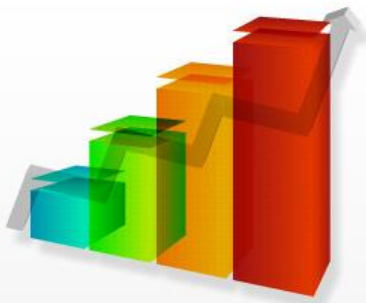
idCliente	NomeCliente	Endereco	Telefones
1	Jose da Silva	Rua Justiano, 3300	2342-2342 / 2242-1210
2	Emanuel Bezerra	Travessa Quinze, 200	8952-2231
3	João dos Santos	Av. Brasil, 220	6758-5391 / 4302-9000 / 7548-9091
4	Maria Mariana	Av. Bandeirantes, 2908	9099-3022

1- Identificação da chave primária da tabela;

2- Identificação da(s) coluna(s) que contem dados multivalorados e removê-las;

3- Criação de uma nova tabela com chave primária para armazenamento do dado repetido;

idClienteTelefone	idCliente	Telefone
1	1	2342-2342
2	1	2232-1210
3	2	8952-2231
4	3	6758-5391
5	3	4302-9000
6	3	7548-9091
7	4	9099-3022



Modelo Lógico

- **Normalização de Dados**

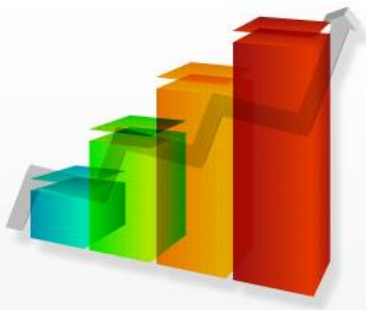
- PRIMEIRA FORMA NORMAL(1FN):

Primeiro exemplo:

4- Criar uma relação entre a tabela principal e a secundária;

idClienteTelefone	idCliente	Telefone
1	1	2342-2342
2	1	2232-1210
3	2	8952-2231
4	3	6758-5391
5	3	4302-9000
6	3	7548-9091
7	4	9099-3022

idCliente	NomeCliente	Endereco
1	Jose da Silva	Rua Justiano, 3300
2	Emanuel Bezerra	Travessa Quinze, 200
3	João dos Santos	Av. Brasil, 220
4	Maria Mariana	Av. Bandeirantes, 2908



Modelo Lógico

- **Normalização de Dados**

- DEPENDÊNCIA FUNCIONAL:

Um campo B possui dependência funcional do campo A se, para cada valor do campo A, existe **exatamente um único** valor do campo B.

A dependência Funcional é representada por $A \rightarrow B$

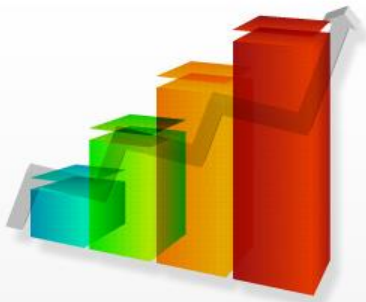
Exemplo:

Cliente	
CPF	Nome
1	José
2	João
3	Rui
4	Manoel

Observe que existe uma dependência entre os valores da tabela, ou seja, se estivermos com o número do CPF, podemos encontrar o nome do cliente correspondente.

Essa dependência é expressa por:

CPF \rightarrow Nome



Modelo Lógico

▪ Normalização de Dados

▪ DEPENDÊNCIA FUNCIONAL: Outras Definições

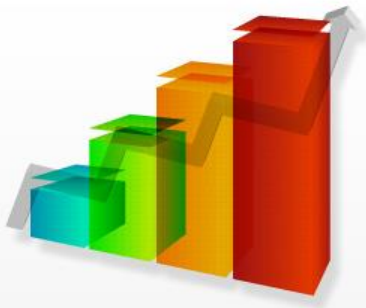
Dados dois conjuntos de campos A e B de uma Tabela, diz-se que:

- B é funcionalmente dependente de A ou
- A determina B ou
- B depende de A.

Se a cada valor de A **estiver associado um, e só um**, valor de B.

- Se **A** determina **B** então **B** não é funcionalmente dependente de nenhum subconjunto de **A**.

$$A \rightarrow B$$



Modelo Lógico

- **Normalização de Dados**

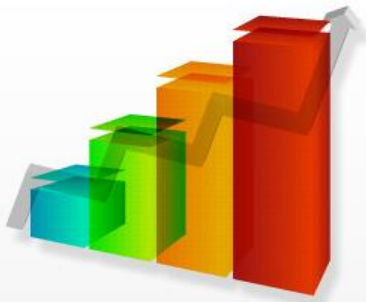
- DEPENDÊNCIA FUNCIONAL: Outras Definições

Outro Exemplo1:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

Departamento → N_funcionário ?

Não pois Departamento 900 => {1021, 1023}



Modelo Lógico

- **Normalização de Dados**

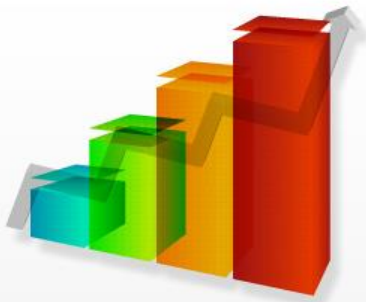
- DEPENDÊNCIA FUNCIONAL: Outras Definições

Outro Exemplo1:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

N_funcionário → Departamento ?

Sim pois se conhecer o N_funcionário (atributo unívoco) é possível determinar o Departamento (um funcionário só pode pertencer a um departamento)



Modelo Lógico

- **Normalização de Dados**

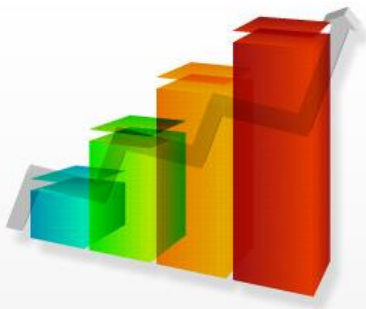
- DEPENDÊNCIA FUNCIONAL: Outras Definições

Outro Exemplo2:

Papelaria	Artigo	Preço
Colmeia	Caneta bic fina	150
Central	Fita cola	300
Aguarela	Borracha	215
Silva	Caneta bic fina	175

A Papelaria é funcionalmente dependente de artigo (Artigo \rightarrow Papelaria) ?

Não; o mesmo artigo pode aparecer em diferentes papelerias



Modelo Lógico

- **Normalização de Dados**

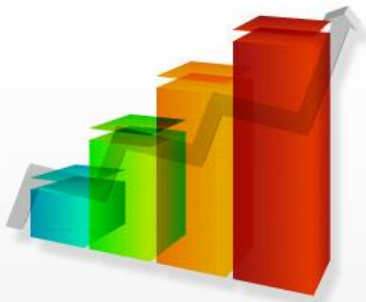
- DEPENDÊNCIA FUNCIONAL: Outras Definições

Outro Exemplo2:

Papelaria	Artigo	Preço
Colmeia	Caneta bic fina	150
Central	Fita cola	300
Aguarela	Borracha	215
Silva	Caneta bic fina	175

O Preço é funcionalmente dependente de Artigo (Artigo \rightarrow Papelaria) ?

Não; pra cada papelaria há tantos valores para preços quantos os artigos vendidos nessa papelaria.



Modelo Lógico

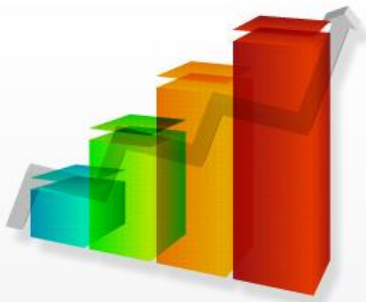
- **Normalização de Dados**

- DEPENDÊNCIA FUNCIONAL: Outras Definições

Outro Exemplo2:

Papelaria	Artigo	Preço
Colmeia	Caneta bic fina	150
Central	Fita cola	300
Aguarela	Borracha	215
Silva	Caneta bic fina	175

Preço depende funcionalmente de ambos {Papelaria, Artigo} → Preço



Modelo Lógico

- **Normalização de Dados**

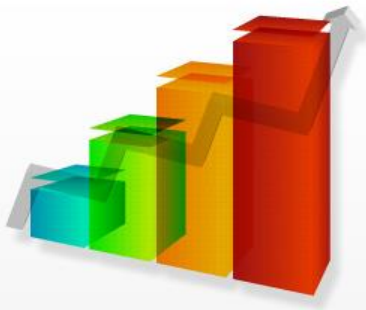
- SEGUNDA FORMA NORMAL(2FN):

Uma Tabela está na **2FN** se e somente estiver na 1FN e não contiver dependências funcionais parciais.

Uma dependência funcional (parcial) ocorre quando uma coluna depende apenas de parte de uma chave primária composta.

Ou seja:

- Uma Tabela é dita na segunda forma normal, se ela estiver na primeira forma normal **E** cada uma das suas colunas não pertencentes à chave primária não for parcialmente dependente desta chave.
- Desta forma, campos que não sejam dependentes exatamente da mesma chave devem ser separados em tabelas diferentes.



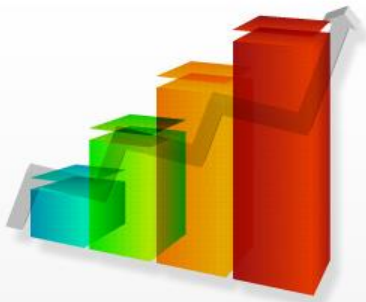
Modelo Lógico

- **Normalização de Dados**

- SEGUNDA FORMA NORMAL(2FN):

Passos:

- Identificar as colunas que não são funcionalmente dependentes da chave primária das tabelas.
- Remover a coluna da tabela e criar uma nova tabela com esses dados.

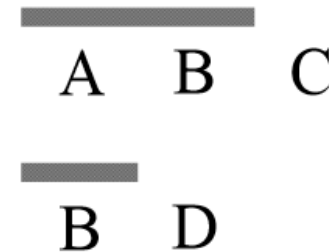
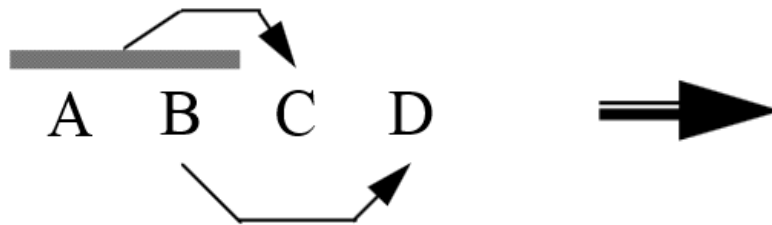


Modelo Lógico

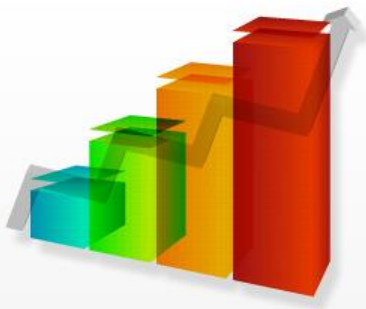
- **Normalização de Dados**

- SEGUNDA FORMA NORMAL(2FN):

Exemplo1:



Portanto, consiste em retirar das estruturas de dados que possuem chaves compostas todos os dados que são funcionalmente dependentes de somente alguma parte dessa chave.



Modelo Lógico

- **Normalização de Dados**

- SEGUNDA FORMA NORMAL(2FN):

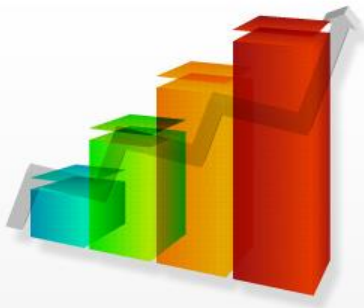
Exemplo2:

TABELA Item de Nota Fiscal

<u>NumNF</u>	<u>CodProd</u>	DescProd	QuantProd	ValorUnit	ValorTot
456123	45	Limpadora a Vácuo	1	600,00	600,00
859032	25	Computador	1	1800,00	1800,00
859631	32	Refrigerador	2	800,00	1600,00
745689	27	Televisão LG	3	950,00	2850,00
485689	45	Limpadora a Vácuo	2	600,00	1200,00

O valor do campo DescProd depende de NumNF , CodProd ?

Não! A descrição do produto depende somente do código do produto. Ou seja, ela varia pela mudança do valor do atributo CodProd somente, assim como ValorUnit que também varia de acordo com o valor de CodProd.



Modelo Lógico

- **Normalização de Dados**

- SEGUNDA FORMA NORMAL(2FN): Exemplo2:

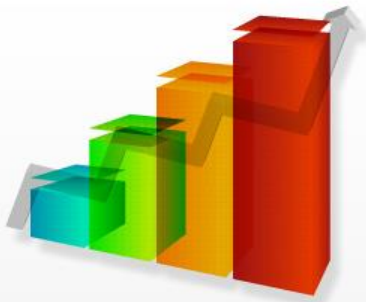
TABELA Item de Nota Fiscal

<u>NumNF</u>	CodProd	QuantProd	ValorTot
456123	45	1	600,00
859032	25	1	1800,00
859631	32	2	1600,00
745689	27	3	2850,00
485689	45	2	1200,00

Resultado das tabelas já na 2FN

TABELA Item de Produtos

<u>CodProd</u>	DescProd	ValorUnit
45	Limpadora a Vácuo	600,00
25	Computador	1800,00
32	Refrigerador	800,00
27	Televisão LG	950,00



Modelo Lógico

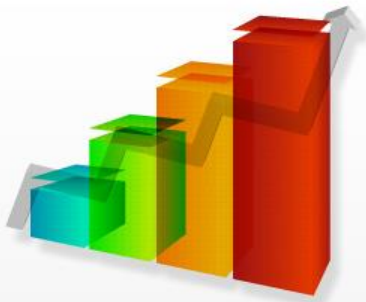
- **Normalização de Dados**

- DEPENDÊNCIA FUNCIONAL TRANSITIVA:

As dependências funcionais podem existir para atributos que não são chaves em uma tabela. Este fato denominamos de **dependência funcional transitiva**.

Exemplo:

<u>Codigo</u>	CampoA	CampoB	CampoC
1	1	1	2
2	1	2	3
3	2	2	4
4	5	5	10
5	10	20	30



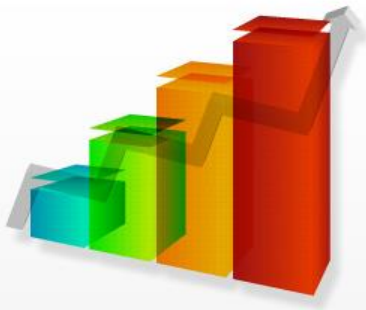
Modelo Lógico

- **Normalização de Dados**

- TERCEIRA FORMA NORMAL(3FN):

Uma Tabela está na **3FN** se e somente estiver na 2FN e não contiver dependências funcionais transitivas.

- Uma tabela é dita na terceira forma normal, se ela estiver na segunda forma normal **E** todos os seus campos só dependem da chave primária.
 - Desta forma, campos que estejam envolvidos numa dependência funcional mútua devem ser transferidos para outra tabela.



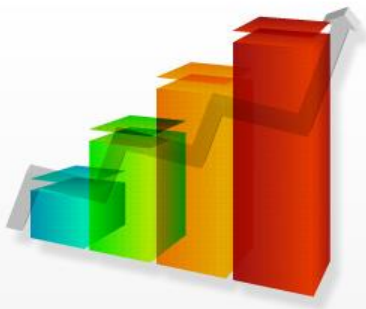
Modelo Lógico

- **Normalização de Dados**

- TERCEIRA FORMA NORMAL(3FN):

Passos:

- Identificar as colunas que são funcionalmente dependentes das outras colunas não chave da tabela.
- Remover essas colunas.



Modelo Lógico

- **Normalização de Dados**

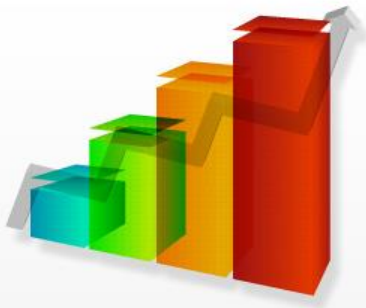
- TERCEIRA FORMA NORMAL(3FN):

Exemplo:

<u>CPF</u>	Nome	Cargo	Salario
111	João	A1	1000,00
112	Maria	A2	2000,00
113	José	A2	2000,00
114	Ana	A3	1500,00

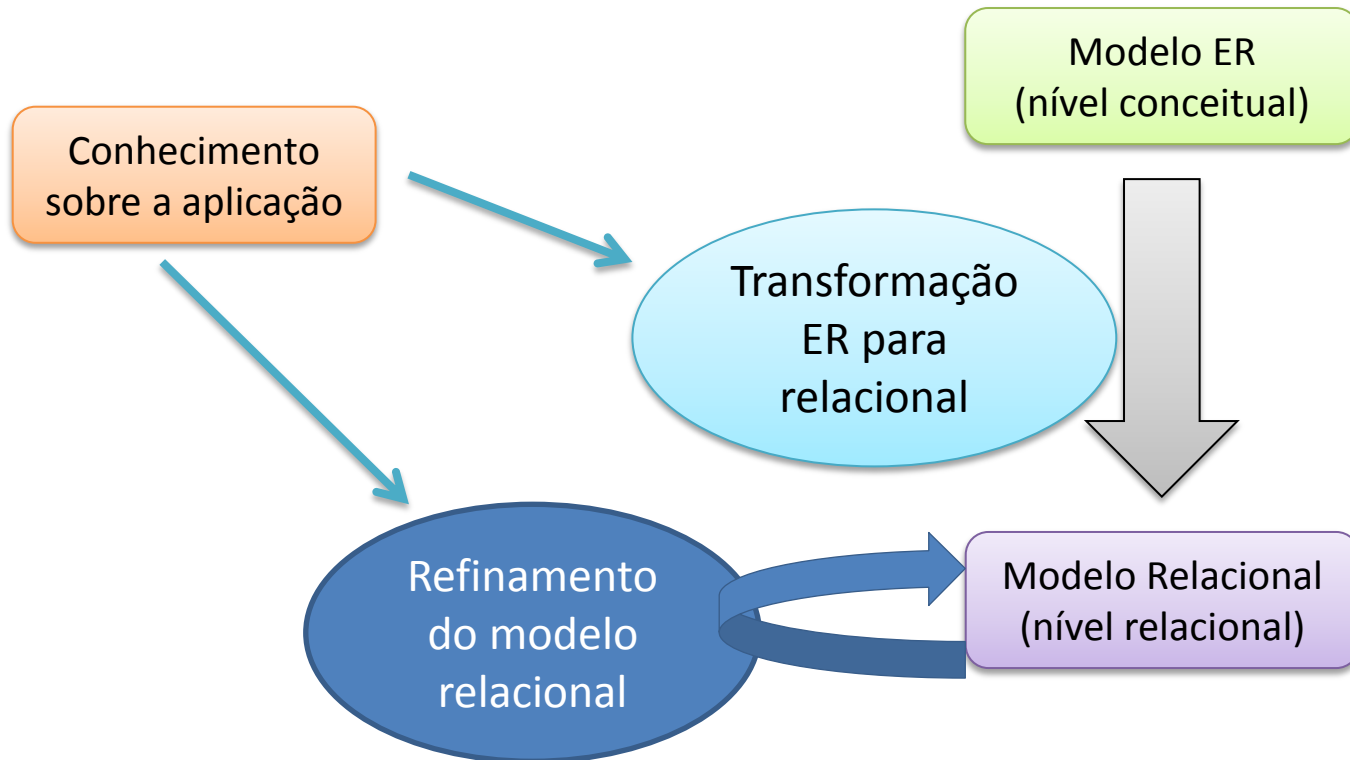
<u>CPF</u>	Nome	Cargo
111	João	A1
112	Maria	A2
113	José	A2
114	Ana	A3

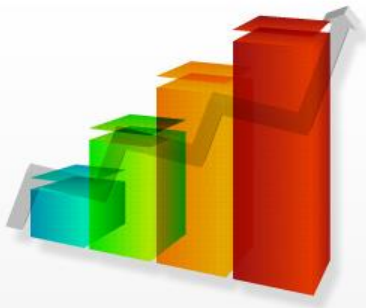
Cargo	Salario
A1	1000,00
A2	2000,00
A2	2000,00
A3	1500,00



Modelo Lógico

- **Transformações Entre Modelos: Regras de Derivação:**





Modelo Lógico

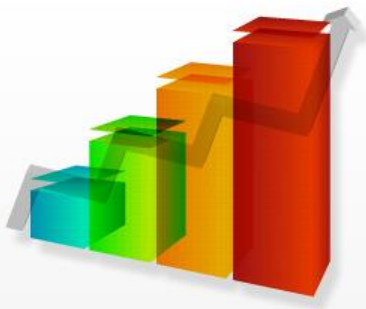
- **Normalização de Dados**

- QUARTA FORMA NORMAL(4FN):

Uma Tabela encontra-se na quarta forma normal, quando, além de estar na 3FN, **não contiver dependência funcional multi-valorada**.

Ou seja:

- Uma tabela é dita na quarta forma normal, se ela estiver na 3FN **E** não tiver nenhum dos seus campos que determinam, com um valor, um conjunto de elementos diferentes.
- Desta forma, campos que estejam envolvidos numa dependência multi-valorada devem ser transferidos para outra tabela.



Modelo Lógico

- Normalização de Dados

- QUARTA FORMA NORMAL (4FN):

3FN: Estantes (número, capacidade)

Livros (ISBN, título, ano)

Autores (codAutor, nome, nacionalidade)

DistribuiçãoLivrosAutor(número, ISBN, codAutor)



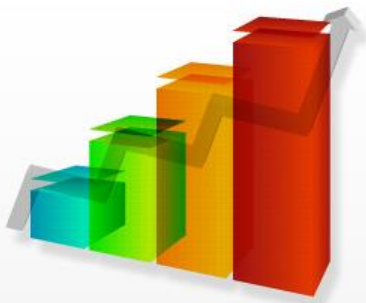
4FN: Estantes (número, capacidade)

Livros (ISBN, título, ano)

Autores (codAutor, nome, nacionalidade)

DistribuiçãoLivros(número, ISBN)

Autoria(ISBN, codAutor)



Modelo Lógico

▪ Normalização de Dados

▪ QUARTA FORMA NORMAL(4FN):

Exemplos - 1:

Autores Assuntos Livros

<u>No tombo</u>	Autor	Assunto
321321-221	Silberschatz. A	Banco de Dados
321321-221	Korth, H.F.	Banco de Dados
321321-221	Silberschatz. A	SQL
321321-221	Korth, H.F.	SQL

No_tombo →→ Autor

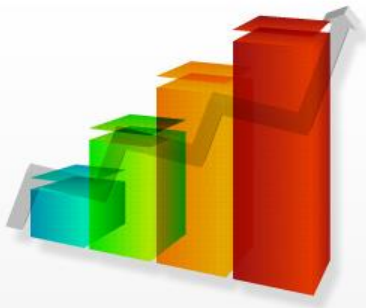
No_tombo →→ Assunto

AutoresLivros

<u>No tombo</u>	Autor
321321-221	Silberschatz. A
321321-221	Korth, H.F.
321321-221	Silberschatz. A
321321-221	Korth, H.F.

AssuntosLivros

<u>No tombo</u>	Assunto
321321-221	Banco de Dados
321321-221	Banco de Dados
321321-221	SQL
321321-221	SQL



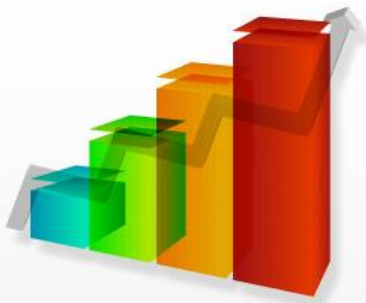
Modelo Lógico

- **Normalização de Dados**

- QUARTA FORMA NORMAL(4FN):

Exemplos - 2:

CodProj	CodEmp	CodEquip
1	1	1
1	2	1
1	3	1
1	1	2
1	2	2
1	3	2
2	2	2
2	2	4
3	3	1
3	4	1
3	3	3
3	4	3
3	3	5



Modelo Lógico

- **Normalização de Dados**

- QUARTA FORMA NORMAL(4FN):

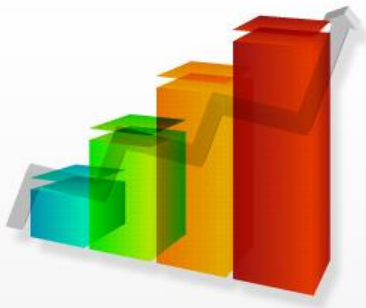
Exemplos - 2:

CodProj	CodEmp	CodEquip
1	1	1
1	2	1
1	3	1
1	1	2
1	2	2
1	3	2
2	2	2

$\text{CodProj} \twoheadrightarrow \text{CodEmp}$

$\text{CodProj} \twoheadrightarrow \text{CodEquip}$

Uma tabela está na 4FN, caso, além de estar na 3FN, não possua mais que uma dependência funcional multi-valorada



Modelo Lógico

- **Normalização de Dados**

- QUARTA FORMA NORMAL(4FN):

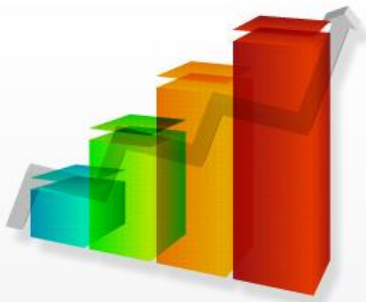
Exemplos - 2:

ProjEmp:

CodProj	CodEmp
1	1
1	2
1	3
2	2
3	3
3	4
4	2

ProjEquip:

CodProj	CodEquip
1	1
1	2
2	2
2	4
3	1
3	3
3	5
4	5



Modelo Lógico

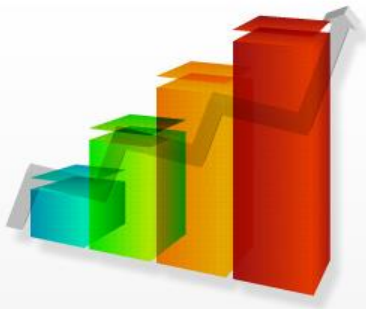
- **Normalização de Dados**

- FORMA NORMAL BOYCE CODD: **BCFN**

Uma Tabela está na **BCFN** quando todos os atributos estão dependentes da chave, de toda a chave e de nada mais do que a chave.

A **BCFN** é um aperfeiçoamento da 3FN, destinada a lidar com situações em que verifique a existência de mais do que uma chave candidata e que duas chaves candidatas possuam elementos comuns.

A **BCFN** foi criada por Raymond Boyce em 1974. Para atender há algumas condições que a 3FN não cobria.



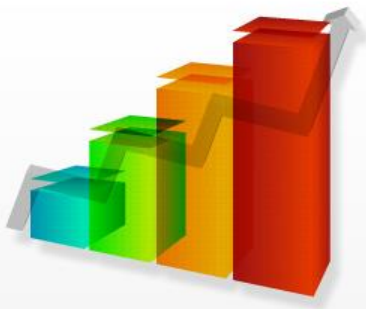
Modelo Lógico

- **Normalização de Dados**

- FORMA NORMAL BOYCE CODD: **BCFN**

As anomalias resolvidas pela **BCFN** aparecem em tabelas com as seguintes características:

- Multiplas chaves candidatas;
- Chaves candidatas são compostas;
- Chaves candidatas que compartilham pelo menos um campo em comum.



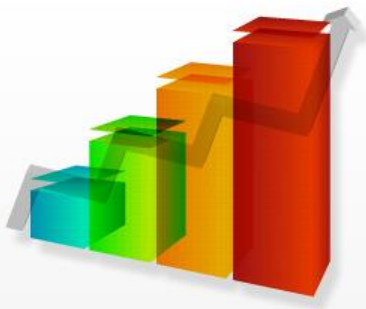
Modelo Lógico

- **Normalização de Dados**
 - FORMA NORMAL BOYCE CODD: BCFN

Exemplo:

CodAluno	CodCurso	CodTurma	CodProfessor
A001	SI01	1P	345
A002	DR01	2P	432
A003	AD01	3P	432

Um mesmo professor pode ministrar aulas entre cursos e turmas diferentes. Sendo assim podemos identificar três chaves candidatas que são determinantes nessa entidade: CODCURSO+CODTURMA, CODCURSO+CODPROFESSOR e CODTURMA+CODPROFESSOR.



Modelo Lógico

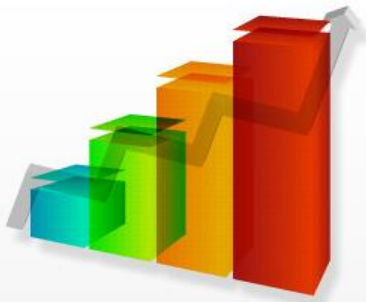
- **Normalização de Dados**
 - FORMA NORMAL BOYCE CODD: BCFN

Exemplo:

CodAluno	CodCurso	CodTurma	CodProfessor
A001	SI01	1P	345
A002	DR01	2P	432
A003	AD01	3P	432

O atributo CODPROFESSOR é parcialmente dependente do CODCURSO e de CODTURMA, mas é totalmente dependente da chave candidata composta CODCURSO+CODTURMA.

Dessa forma a entidade deve ser desmembrada, resultando em duas: uma que contem os atributos que descrevem o aluno em si, e outra cujos atributos designam um professor.



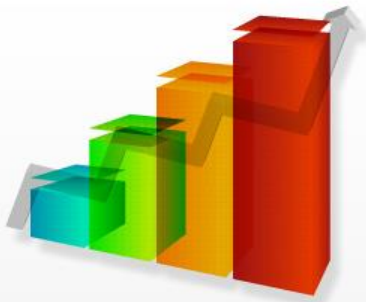
Modelo Lógico

- **Normalização de Dados**
 - FORMA NORMAL BOYCE CODD: BCFN

Exemplo:

CodAluno	CodCurso	CodTurma
A001	SI01	1P
A002	DR01	2P
A003	AD01	3P

CodCurso	CodTurma	CodProfessor
SI01	1P	345
DR01	2P	432
AD01	3P	432

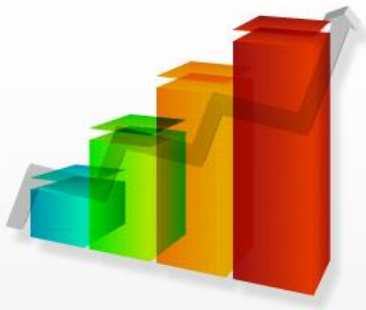


Modelo Lógico

- **Normalização de Dados**

- QUINTA FORMA NORMAL: 5FN

- Uma relação de 4FN estará em 5FN, quando seu conteúdo não puder ser reconstruído (existir perda de informação) a partir das diversas relações menores que não possuam a mesma chave primária.
- A 5FN trata de casos bastante particulares, que ocorrem na modelagem de dados, que são os relacionamentos múltiplos (ternários, quaternários e n-ários).
- Ela fala que um registro está na sua 5FN, quando o conteúdo deste mesmo registro não puder ser reconstruído (junção) a partir de outros registros menores, extraídos deste registro principal. Ou seja, se ao particionar um registro, e sua junção posterior não conseguir recuperar as informações contidas no registro original, então este registro está na 5FN.

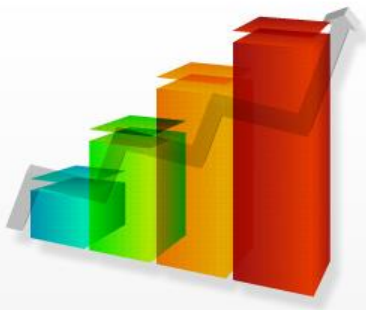


Modelo Lógico

- **Normalização de Dados**

- Benefícios:

- Estabilidade do Modelo Lógico
Ex: Itens sem repetição
 - Integridade
 - Economia
Ex: Custo de manipulação de dados. Backup, restauração, atualização, etc
 - Fidelidade ao ambiente observado



Modelagem de Dados

Referências

DATE, C. J. **Introdução a Sistemas de Banco de Dados**. 8ª Edição, São Paulo, Campus, 2004.

ELMASRI, Ramez E.; NAVATHE, Shamkant. **Sistemas de Banco de Dados: Fundamentos e Aplicações**. 6ª Edição, São Paulo, Addison Wesley, 2011.

MACHADO, Felipe Nery Rodrigues. **Banco de Dados - Projeto e Implementação**. 2ª Edição, São Paulo, Editora Érica, 2011.

HEUSER, Carlos Alberto. **Projeto de Banco de Dados**. 6a ed. Porto Alegre, 2009.

CHEN, Peter. **Modelagem de Dados: a abordagem entidade-relacionamento para projeto lógico**. Trad. Cecília Camargo Bartalotti. São Paulo: Makron Books, 1990.