



**PONTIFÍCIA UNIVERSIDADE CATÓLICA GOIÁS
CURSO DE ENGENHARIA DE COMPUTAÇÃO
ESCOLA DE CIÊNCIAS EXATAS E DA COMPUTAÇÃO**

NORMALIZAÇÃO DE DOCUMENTOS

GOIÂNIA 2017

PONTIFÍCIA UNIVERSIDADE CATÓLICA GOIÁS

NORMALIZAÇÃO DE DOCUMENTOS

LUCAS MACEDO DA SILVA

GOIÂNIA, 2017

PONTIFÍCIA UNIVERSIDADE CATÓLICA GOIÁS

MODELO ENTIDADE RELACIONAMENTO- MER

Trabalho apresentado por Lucas Macedo da Silva e, para avaliação e aplicação dos conceitos aprendidos em sala de aula.

GOIÂNIA, 2017

Resumo

Os documentos de dados servem como base para se modelar um banco de dados relacional, onde o usuário tende a fornecer, algum arquivo que consta os dados que até agora o mesmo possui, bem como são armazenados, isso implica em um documento não normalizado, o que muitas vezes feri vários critérios da abordagem relacional de um banco de dados, cabe então ao responsável para a tradução normaliza – ló, o que implica na passagem das três formas normais, onde a primeira busca eliminar tabelas multivaloradas, a segunda eliminar dependências com a chave primária e a terceira eliminar dependências com colunas que não fazem parte da chave primária, de forma que ao final do processo, o modelo logico resultante pode ser aplicado a abordagem relacional não gerando inconsistência de dados, bem como redundâncias.

Abstract

The data documents serve as a basis for modeling a relational database, where the user has tended to provide, some file that contains the data that he has so far and are stored, this implies in a non-standard document, the which often hurt several criteria of the relational approach of a database, it is then up to the person responsible for the normalized translation, which implies the passage of the three normal forms, where the first search eliminates multivalued tables, the second eliminates dependencies with the primary key and the third eliminate dependencies with columns that are not part of the primary key, so that at the end of the process, the resulting logical model can be applied to the relational approach not generating data inconsistency as well as redundancies.

Lista de Figuras

Figura 01 – Modelo Entidade Relacionamento.

12

Glossário de Siglas

CPF	Cadastro de Pessoa Física
ÑN	Forma não normalizada
1FN	Primeira Forma Normal
2FN	Segunda Forma Normal
3FN	Terceira Forma Normal
Cod	Código
SQL	Linguagem de Consulta Estruturada

Sumário

Sumário	8
Introdução.....	9
Forma Não Normalizada (ÑN)	10
Primeira Forma Normal (1FN).....	10
Segunda Forma Normal (2FN).....	10
Terceira Forma Normal (3FN).....	11
Modelo Entidade Relacionamento	12
Tradução Modelo Entidade Relacionamento.....	12
Tradução Relacionamento FaturaCliente	12
Tradução Relacionamento CarroCategoria	13
Tradução Relacionamento FaturaCarro	13
Tradução do Relacionamento Possui.....	14
Tradução Relacionamento Pertenci.....	14
Solução Final	15
Modelo em SQL.....	16
Cliente	16
Marca	16
Modelo.....	16
Carro	16
Categoria.....	17
Fatura	17
FaturaCarro	17
Conclusão.....	19
Referências Bibliográficas.....	20
Apêndice.....	21
Inserções:	21
Anexo A	24
Tabela Não Normalizada	24

Introdução

A representação dos dados de um banco de dados, comumente se dá a partir de documentos não normalizados, ou seja, documentos que apresentam estruturas inexistentes na abordagem relacional a principal dela são colunas multivaloradas, a normalização serve para contornar esse problema e tornar possível a implementação do banco de dados, evitando redundância de dados e onde as colunas de uma tabela são todas atômicas, os critérios a serem seguidos durante a atividade de normalização permitem isso, o último estágio constrói o modelo lógico que não gera nenhuma inconsistência de dados e que utiliza apenas estruturas existentes na abordagem relacional.

Forma Não Normalizada (ÑN)

Na forma não normalizada a tabela apresenta colunas aninhadas, com isso apresenta uma estrutura inexistente em um banco de dados relacional.

Com isso a tabela apresentada, possui a seguinte forma não normalizada:

Relatório (CPE, nome, tipo, (codFatura, qtdDiarias, KMinicial, KMFinal, Placa, codModelo, codCategoria, valorUmaDiaria, codMarca).

Primeira Forma Normal (1FN)

Na primeira forma normal a tabela não apresenta colunas aninhadas, eliminando assim o problema de a normalização gerar estruturas inexistentes na abordagem relacional.

Com isso as tabelas apresentadas, apresentam a seguinte primeira forma normal:

Cliente (CPE, nome, tipo).

Fatura (codFatura, CPF, Placa, codModelo, codMarca, codCategoria, valorUmaDiaria, KMinicial, KMFinal, qtdDiarias).

CPF referencia Cliente.

Segunda Forma Normal (2FN)

Na segunda forma normal a tabela já está na 1FN e não apresenta dependências parciais, ou seja, cada coluna não chave depende da chave primária completa.

Com isso as tabelas apresentadas, possuem a seguinte segunda forma normal:

Cliente (CPE, nome, tipo).

Fatura (codFatura, CPFCliente).

CPFCliente referencia Cliente.

FaturaCarro (codFatura, Placa, KMinicial, KMFinal, qtdDiarias).

codFatura referencia Fatura.

Placa referencia Carro.

Carro (Placa, codModelo, codMarca, codCategoria, valorUmaDiaria).

Terceira Forma Normal (3FN)

Na terceira forma normal a tabela já está na 2FN e não apresenta dependências transitivas, ou seja, não apresenta colunas não chave primária que dependem funcionalmente de outra coluna ou combinação de colunas não chave primária.

Com isso as tabelas apresentadas, possuem a seguinte terceira forma normal:

Cliente (CPF, nome, tipo).

Fatura (codFatura, CPFCliente).

CPFCliente referencia Cliente.

Categoria (codCategoria, valorUmaDiaria).

FaturaCarro (codFatura, Placa, KMInicial, KMFinal, qtdDiarias).

codFatura referencia Fatura.

Placa referencia Carro.

Carro (Placa, codMarca, codCategoria).

codMarca referencia marca.

codCategoria referencia Categoria.

Modelo (codModelo, codMarca).

codMarca referencia Marca.

Marca (codMarca, descrMarca).

Modelo Entidade Relacionamento

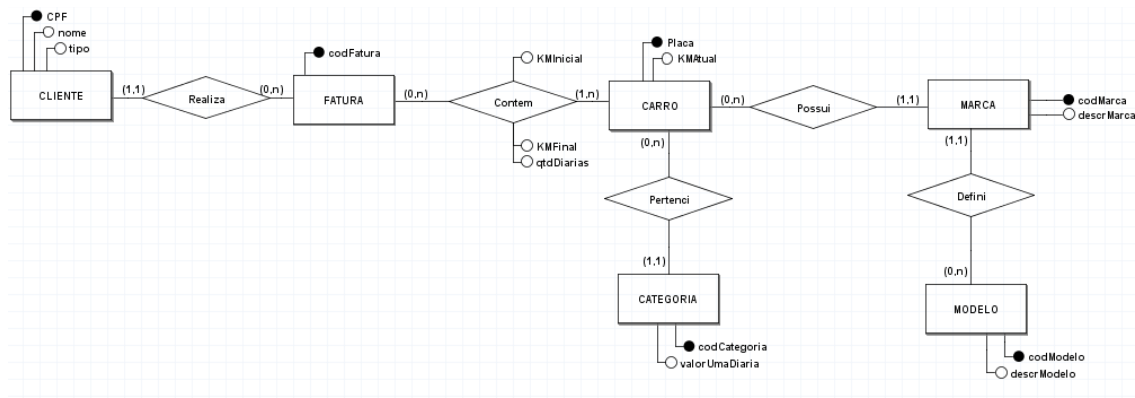


Figura 01 – Modelo Entidade Relacionamento.

Tradução Modelo Entidade Relacionamento

Cliente (CPF, nome, tipo).

Fatura (codFatura).

Categoria (codCategoria, valorUmaDiaria).

Carro (Placa, KMAtual).

Contem (KMinicial, KMFinal, qtdDiarias).

Modelo (codModelo, descrModelo).

Marca (codMarca, descrMarca).

Tradução Relacionamento Realiza

Temos um relacionamento 1:n, onde a solução preferível para a tradução é a partir de adição de colunas, já que a partir dela diminuimos o número de junções, os campos opcionais são aqueles referentes a atributos opcionais, bem como diminuimos o uso de chaves já que a chave primária da tabela é a apenas uma, o que reduz o número de índices gerados pelo banco de dados aumentando assim a performance do mesmo, já a tradução a partir de fusão de tabelas não se aplica por gerar colunas

aninhadas, o que retornaria o modelo para a forma não normalizada, vale ressaltar que na abordagem relacional existem apenas colunas atômicas e essa tradução, faz com que crie-se uma estrutura não existentes na abordagem, já a tradução a partir de tabela única também pode ser aplica porém geraria maior índice de junções e uso de chaves o que influencia na performance do banco, porém não apresentaria problemas com atributos opcionais.

Com isso a alternativa escolhida é a de tradução a partir de adição de colunas, com isso temos a seguinte tradução:

Cliente (CPF, nome, tipo).

Fatura (codFatura, CPFCliente).

CPFCliente referencia Cliente.

Tradução Relacionamento Pertenci

Por se tratar de um relacionamento com cardinalidades máxima 1:n, a alternativa preferível para a tradução do relacionamento é a adição de colunas, já que o problema com campos opcionais está relacionado apenas com campos opcionais, apresenta menor número de junções e uso de colunas para compor a chave primária da tabela, já a tradução a partir de tabela própria pode ser aplicada como uma segunda opção caso não se opte pela primeira, porém apresentam maior número de junções e uso de chaves primárias, já a alternativa por tabela própria não se aplica pois pode gerar inconsistência de dados.

Portanto a partir de adição de colunas temos a seguinte tradução do relacionamento:

Categoria (codCategoria, valorUmaDiaria).

Carro (Placa, codCategoria, KMAtual).

codCategoria referencia Categoria.

Tradução Relacionamento Contem

Em relacionamentos n:n, a alternativa preferível é a tabela própria, já que é a única das possíveis alternativas que permite o fácil gerenciamento dos dados, mantendo as restrições, bem como as colunas continuam sendo atômicas que são estruturas existentes na abordagem relacional, já a fusão de tabelas gera um alto nível de redundância de dados, que pode não ser bom para o banco, e adição de colunas pode gerar colunas aninhadas.

Portanto a partir de tabela própria temos a seguinte tradução do relacionamento:

Fatura (codFatura, CPFCliente).

CPFCliente referencia Cliente.

Carro (Placa, codCategoria, KMAtual).

FaturaCarro (codFatura, Placa, KMinicial, KMFinal, qtdDiarias).

codFatura referencia Fatura

Placa referencia Carro.

Tradução do Relacionamento Possui

Em relacionamentos do tipo 1:n, a solução preferível para a tradução é a partir de adição de colunas, já que assim, temos menor número de junções, pois os dados serão buscados em apenas duas tabelas, os campos opcionais são aqueles referentes a atributos opcionais, bem como temos o menor uso de chaves diminuindo o número de índices criados pelo banco de dados e consequentemente aumentando a performance do mesmo, já a tradução a partir de tabela própria pode ser aplicada como uma segunda alternativa caso a primeira não caiba, os campos opcionais são apenas aqueles inerentes a atributos opcionais, porém apresenta alto índice de junções já que os dados terão de ser buscados em três tabelas diferentes não apenas em duas como na adição de colunas, bem como há um maior uso de chaves primárias, o que aumenta o número de índices gerados pelo banco para controle da unicidade das chaves primarias, diminuindo assim a performance do banco, a solução a partir de fusão de tabelas não se aplica por gerar muitas redundâncias e colunas aninhadas que não existem na abordagem relacional.

Portanto utilizando adição de colunas obtemos a seguinte tradução:

Carro (Placa, codCategoria, KMAtual, codMarca).

codMarca referencia marca.

Marca (codMarca, descrMarca).

Tradução Relacionamento Pertenci

Trata-se de um relacionamento de cardinalidades máximas 1:n, logo a alternativa preferível é a de adição de colunas, pois assim há menor uso de colunas para compor

a chave primária da tabela, o problema com campos opcionais está relacionados apenas a atributos opcionais, há também menor número de junções, já a tradução por tabela própria pode ser aplicada como uma segunda opção caso a primeira não caiba, não apresenta problemas com campos opcionais, porém gera maior uso de junções e chaves para compor a chave primária, tornando a performance do banco mais lenta, já a tradução por fusão de tabelas não se aplica a este tipo de relacionamento.

Portanto optando por adição de colunas temos a seguinte tradução:

Marca (codMarca, descrMarca).

Modelo (codModelo, descrModelo, codMarca).

codMarca referencia Marca.

Solução Final

Ao final da discussão obtemos a seguinte tradução para o modelo entidade relacionamento da figura 01:

Cliente (CPE, nome, tipo).

Marca (codMarca, descrMarca).

Modelo (codModelo, descrModelo, codMarca).

codMarca referencia Marca.

Carro (Placa, codMarca, codCategoria, KMAtual).

codMarca referencia marca.

codCategoria referencia Categoria.

Fatura (codFatura, CPFCliente).

CPFCliente referencia Cliente.

FaturaCarro (codFatura, Placa, KMinicial, KMFinal, qtdDiarias).

codFatura referencia Fatura.

Placa referencia Carro.

Categoria (codCategoria, valorUmaDiaria).

Modelo em SQL

Cliente

```
CREATE TABLE Cliente (  
    CPF DECIMAL (11) PRIMARY KEY,  
    nome VARCHAR (50) NOT NULL,  
    tipo VARCHAR (30) NOT NULL  
);
```

Marca

```
CREATE TABLE Marca (  
    codMarca INTEGER PRIMARY KEY,  
    descMarca VARCHAR (30) NOT NULL UNIQUE  
);
```

Modelo

```
CREATE TABLE Modelo (  
    codModelo INTEGER PRIMARY KEY,  
    descModelo VARCHAR (30) NOT NULL UNIQUE,  
    codigoMarca INTEGER NOT NULL,  
    FOREIGN KEY (codigoMarca) REFERENCES Marca (codMarca)  
);
```

Carro

```
CREATE TABLE Carro (  
    placa VARCHAR (8) PRIMARY KEY,
```



```

codigoMarca INTEGER NOT NULL,
codigoCategoria VARCHAR (30) NOT NULL,
codigoModelo INTEGER NOT NULL,
KMAtual REAL NOT NULL CHECK (KMAtual >= 0),
FOREIGN KEY (codigoModelo) REFERENCES Modelo (codModelo),
FOREIGN KEY (codigoMarca) REFERENCES Marca (codMarca),
FOREIGN KEY (codigoCategoria) REFERENCES Categoria (codCategoria)
);

```

Categoria

```

CREATE TABLE Categoria (
    codCategoria VARCHAR (30) PRIMARY KEY,
    valorUmaDiaria REAL NOT NULL CHECK (valorUmaDiaria >= 0)
);

```

Fatura

```

CREATE TABLE Fatura (
    codFatura INTEGER PRIMARY KEY,
    CPFCliente DECIMAL (11) NOT NULL,
    FOREIGN KEY (CPFCliente) REFERENCES Cliente (CPF)
);

```

FaturaCarro

```

CREATE TABLE FaturaCarro (
    codigoFatura INTEGER NOT NULL,
    placa VARCHAR (8) NOT NULL,
    FOREIGN KEY (placa) REFERENCES Carro (placa),
    FOREIGN KEY (codigoFatura) REFERENCES Fatura (codFatura),
    KMInicial INTEGER NOT NULL CHECK (KMInicial >= 0),
    KMFinal INTEGER NOT NULL CHECK (KMFinal >= 0),

```

```
qtdDiarias INTEGER NOT NULL CHECK (qtdDiarias >= 0),  
PRIMARY KEY (placa, codigoFatura)  
);
```

Conclusão

Portanto em um projeto de banco de dados, comumente o usuário apresenta um documento como base para a modelagem do banco, esse documento costuma não estar normalizado, cabe então a aplicação das técnicas de normalização que buscam criar um banco que esteja de acordo com os requisitos, com a abordagem relacional, que permite armazenar os dados de forma consistente e evite ao máximo redundância de dados, o último estágio tendi a traduzir o documento de forma que atenda tais fatores.

Referências Bibliográficas

[1] Heuser, Carlos Alberto Heuser, Projeto de Banco de Dados, 4º Edição, Instituto de Informática da UFRGS, Sagra, 1998.

Apêndice

Inserções:

Cliente

```
INSERT INTO Cliente VALUES (54367893211, 'ANTONIO CARLOS MENDES', 'RUBY');
```

```
INSERT INTO Cliente VALUES (11239874566, 'MARIA SILVA CARVALHO', 'RUBY');
```

```
INSERT INTO Cliente VALUES (98765432122, 'CARLOS DA SILVA MENDES', 'DIAMANTE');
```

```
INSERT INTO Cliente VALUES (12345678900, 'JOSÉ ANTONIO DA SILVA', 'VIP');
```

Marca

```
INSERT INTO Marca VALUES (1, 'VOLKSWAGEN');
```

```
INSERT INTO Marca VALUES (2, 'GM');
```

```
INSERT INTO Marca VALUES (3, 'TOYOTA');
```

```
INSERT INTO Marca VALUES (4, 'MERCEDES');
```

```
INSERT INTO Marca VALUES (5, 'VOLVO');
```

Modelo

```
INSERT INTO Modelo VALUES (1, 'GOL', 1);
```

```
INSERT INTO Modelo VALUES (2, 'COBALT', 2);
```

```
INSERT INTO Modelo VALUES (3, 'CAPTIVA', 2);
```

```
INSERT INTO Modelo VALUES (4, 'SW4', 3);
```

```
INSERT INTO Modelo VALUES (5, 'C-18', 4);
```

```
INSERT INTO Modelo VALUES (6, 'V60', 5);
```

Carro

```
INSERT INTO Carro VALUES ('ABC-1234', 1, 'ECÔNOMICO', 1, 1700);
```

```
INSERT INTO Carro VALUES ('JCW-4321', 2, 'INTERMEDIÁRIO', 2, 14560);
```

```

INSERT INTO Carro VALUES ('ASD-3456', 2, 'SUV', 3, 1590);
INSERT INTO Carro VALUES ('WER-5678', 3, 'SUV', 4, 6570);
INSERT INTO Carro VALUES ('QWS-1234', 4, 'EXECUTIVO', 5, 1350);
INSERT INTO Carro VALUES ('QAS-1256', 5, 'EXECUTIVO', 6, 1500);
INSERT INTO Carro VALUES ('KOP-1256', 5, 'EXECUTIVO', 6, 45500);
INSERT INTO Carro VALUES ('RNU-3456', 2, 'INTERMEDIÁRIO', 2, 41590);
INSERT INTO Carro VALUES ('TRB-1234', 4, 'EXECUTIVO', 5, 41350);

```

Categoria

```

INSERT INTO Categoria VALUES ('ECÔNOMICO', 65);
INSERT INTO Categoria VALUES ('INTERMEDIÁRIO', 75);
INSERT INTO Categoria VALUES ('SUV', 110);
INSERT INTO Categoria VALUES ('EXECUTIVO', 250);

```

Fatura

```

INSERT INTO Fatura VALUES (1230, 12345678900);
INSERT INTO Fatura VALUES (1330, 12345678900);
INSERT INTO Fatura VALUES (3450, 12345678900);
INSERT INTO Fatura VALUES (235, 12345678900);
INSERT INTO Fatura VALUES (987, 98765432122);
INSERT INTO Fatura VALUES (569, 98765432122);
INSERT INTO Fatura VALUES (2987, 11239874566);
INSERT INTO Fatura VALUES (23456, 11239874566);
INSERT INTO Fatura VALUES (12346, 54367893211);
INSERT INTO Fatura VALUES (56435, 54367893211);

```

FaturaCarro

```

INSERT INTO FaturaCarro VALUES (1230, 'ABC-1234', 1000, 1700, 5);
INSERT INTO FaturaCarro VALUES (1330, 'JCW-4321', 13450, 14560, 8);

```

```
INSERT INTO FaturaCarro VALUES (3450, 'ASD-3456', 23459, 28450, 4);
INSERT INTO FaturaCarro VALUES (3450, 'WER-5678', 4578, 6570, 6);
INSERT INTO FaturaCarro VALUES (235, 'QWS-1234', 3240, 5689, 7);
INSERT INTO FaturaCarro VALUES (987, 'QAS-1256', 1000, 1500, 6);
INSERT INTO FaturaCarro VALUES (987, 'QWS-1234', 1000, 1350, 6);
INSERT INTO FaturaCarro VALUES (569, 'ASD-3456', 1290, 1590, 3);
INSERT INTO FaturaCarro VALUES (2987, 'KOP-1256', 21350, 25500, 8);
INSERT INTO FaturaCarro VALUES (2987, 'RNU-3456', 21290, 21590, 5);
INSERT INTO FaturaCarro VALUES (23456, 'TRB-1234', 21000, 21350, 7);
INSERT INTO FaturaCarro VALUES (12346, 'KOP-1256', 31350, 45500, 18);
INSERT INTO FaturaCarro VALUES (12346, 'RNU-3456', 31290, 41590, 15);
INSERT INTO FaturaCarro VALUES (56435, 'TRB-1234', 31000, 41350, 17);
```

Anexo A

Tabela Não Normalizada

CPF: 123456789				NOME: JOSÉ ANTONIO DA SILVA				TIPO: VIP		
PLACA	FATURA	MODELO	CATEGORIA	DIÁRIA	KM INICIAL	KM FINAL	MARCA	QUANTIDADE DE DIÁRIAS	VALOR DIARIAS	VALOR PAGO
ABC-1234	1230	GOL	ECÔNOMICO	R\$65,00	1000	1700	VOLKSWAGEN	5	R\$325,00	R\$325,00
JCV-4321	1330	COBALT	INTERMEDIÁRIO	R\$75,00	13450	14560	GM	8	R\$600,00	R\$600,00
ASD-3456	3450	CAPTIVA	SUV	R\$110,00	23459	28450	GM	4	R\$440,00	-
VER-5678	3450	SW4	SUV	R\$110,00	4578	6570	TOYOTA	6	R\$660,00	R\$1100,00
QVS-1234	235	C-18	EXECUTIVO	R\$250,00	3240	5689	MERCEDES	7	R\$1750,00	R\$1750,00
CPF: 987654321				NOME: CARLOS DA SILVA MENDES				TIPO: DIAMANTE		
PLACA	FATURA	MODELO	CATEGORIA	DIÁRIA	KM INICIAL	KM FINAL	MARCA	QUANTIDADE DE DIÁRIAS	VALOR TOTAL	VALOR PAGO
QAS-1256	987	V60	EXECUTIVO	R\$250,00	1000	1500	VOLVO	6	R\$1500,00	-
QVS-1234	987	C-18	EXECUTIVO	R\$250,00	1000	1350	MERCEDES	4	R\$1000,00	R\$2500,00
ASD-3456	569	CAPTIVA	SUV	R\$110,00	1290	1590	GM	3	R\$330,00	R\$330,00
CPF: 1123987456				NOME: MARIA SILVA CARYALHO				TIPO: RUBY		
PLACA	FATURA	MODELO	CATEGORIA	DIÁRIA	KM INICIAL	KM FINAL	MARCA	QUANTIDADE DE DIÁRIAS	VALOR TOTAL	VALOR PAGO
KOP-1256	2987	V60	EXECUTIVO	R\$250,00	21350	25500	VOLVO	8	R\$2000,00	-
RNU-3456	2987	COBALT	INTERMEDIÁRIO	R\$75,00	21290	21590	GM	5	R\$375,00	R\$2375,00
TRB-1234	23456	C-18	EXECUTIVO	R\$250,00	21000	21350	MERCEDES	7	R\$1750,00	R\$1750,00
									0	
CPF: 5436789321				NOME: ANTONIO CARLOS MENDES				TIPO: RUBY		
PLACA	FATURA	MODELO	CATEGORIA	DIÁRIA	KM INICIAL	KM FINAL	MARCA	QUANTIDADE DE DIÁRIAS	VALOR TOTAL	VALOR PAGO
KOP-1256	12987	V60	EXECUTIVO	R\$250,00	31350	45500	VOLVO	18	R\$4500,00	-
RNU-3456	12346	COBALT	INTERMEDIÁRIO	R\$75,00	31290	41590	GM	15	R\$1125,00	R\$5625,00
TRB-1234	56435	C-18	EXECUTIVO	R\$250,00	31000	41350	MERCEDES	17	R\$4250,00	R\$4250,00