



Universidade Federal do Ceará
Campus Crateús

Fundamentos de Banco de Dados

Aula 05 - Dependência Funcional e Normalização



Professora Vitória Regina - vitoria@crateus.ufc.br



O que estudaremos?

- Dependência funcional.
- Normalização das relações.
- O grau de normalização.



Introdução

- A qualidade de um projeto de BD pode ser mensurada através de três atributos:
 - Semântica dos atributos;
 - Redução de valores redundantes nas tuplas;
 - Redução de valores nulos nas tuplas.



Introdução

- **Semântica dos atributos da relação:**

- Todo atributo significa alguma informação no mundo real;
- A facilidade com o que o significado dos atributos pode ser entendido é uma medida de um bom projeto de BD.

Introdução

■ Semântica dos atributos da relação:

- Algumas ações ajudam a manter clara a semântica dos atributos das relações:
 - Usar bons nomes para os atributos;
 - Não misturar os dados de entidades distintas em uma única relação.

Introdução

- **Informações redundantes nas tuplas:**

- Um bom projeto de BD visa minimizar o espaço de armazenamento dos dados;
- Isto pode ser feito evitando a redundância de informações nas tuplas.

Introdução

■ Informações redundantes nas tuplas:

- Vamos supor a seguinte tabela:

Matrícula	Nome	Salário	CodDept o	NomeDept o	Gerent e
1111-1	João	2500	1	Vendas	1111-4
1111-2	Maria	2500	2	Financeiro	1111-3
1111-3	Carlos	4500	2	Financeiro	1111-3
1111-4	Joaquim	4500	1	Vendas	1111-4
1111-5	Marcos	4500	3	Jurídico	1111-5

Introdução

■ Informações redundantes nas tuplas:

- Bancos de dados mal projetados com muita redundância de informações podem levar aos seguintes problemas:
 - Anomalias de inserção:
 - Como inserir um novo empregado ou um novo departamento?
 - Anomalias de exclusão:
 - O que acontece se o único empregado de um departamento for excluído?
 - Anomalias de atualização:
 - O que acontece se mudar o gerente de um departamento?

Introdução

■ Valores nulos em tuplas:

- Alguns BD's podem criar relações grandes, com muitos atributos que não se aplicam a todas as tuplas da relação.
- Resultado: Muitas tuplas com valores nulos.
- Valores nulos podem ser interpretados como:
 - O atributo não se aplica à tupla;
 - O valor existe mas é desconhecido.

Introdução

■ Valores nulos em tuplas:

- Vamos observar a seguinte tabela:

Matrícula	Nome	Salário	GerenciaDepartamento
1111-1	João	2500	NULL
1111-2	Maria	2500	NULL
1111-3	Carlos	4500	2
1111-4	Joaquim	4500	1
1111-5	Marcos	4500	3
1111-6	Ana	2500	NULL
1111-7	Luís	2500	NULL

Introdução

■ Valores nulos em tuplas:

- Ter muitas tuplas com valores nulos pode levar aos seguintes problemas:
 - Desperdício de espaço;
 - Problemas no entendimento do significado do atributo;
 - Problemas com operações de junção.

Dependência Funcional

- Uma dependência funcional é uma restrição entre dois conjuntos de atributos.
- Representada da forma $X \longrightarrow Y$;
- Onde:
 - X e Y são dois conjuntos de atributos;
 - Dizemos que o conjunto X determina funcionalmente o conjunto Y e que Y é funcionalmente dependente de X.
- Significa que, para cada conjunto de valores de X, só existe uma combinação de valores de Y.

Dependência Funcional

■ Exemplos:

- Matrícula \longrightarrow {Nome, Salário, Supervisor, CodDepartamento}
- CodDepartamento \longrightarrow {Nome, Gerente}
- {Empregado, CodProjeto} \longrightarrow NumHoras

Dependência Funcional

- Uma dependência funcional é total se a remoção de qualquer atributo de X implica que a dependência não será mais assegurada;
 - **Exemplo:** {Empregado, CodProjeto} \longrightarrow NumHoras é uma dependência funcional total:
 - A remoção de algum dos atributos implica na perda da dependência.
 - **Exemplo:** {Matrícula, Nome} \longrightarrow Salário não é uma dependência funcional total:
 - A remoção do atributo nome mantém a dependência funcional.

Normalização de relações

- Conjunto de testes para avaliar a qualidade de uma relação;
- Para cada avaliação, são feitos testes para verificar se elas atendem às formas normais;
- Foi introduzida por Codd, que definiu **três formas normais** a que as relações devem obedecer.

Primeira Forma Normal

- Uma relação está em **1FN** se ela contém apenas atributos atômicos;
- Isso implica que:
 - Cada atributo da relação é indivisível;
 - Um atributo não pode ter um conjunto de valores;
 - Não existe nenhum atributo multivalorado dentro da tabela.

Primeira Forma Normal

- **Exemplo:** Esta tabela não está na 1FN;
 - O atributo Telefones é multivalorado:

Matrícula	Nome	Salario	Telefones
1111-1	João	2500,00	12341234, 98762334
1111-2	Maria	2500,00	12585897, 12547895
1111-3	Carlos	4500,00	12348792
1111-4	Joaquim	4500,00	12345874

Primeira Forma Normal

- **Exemplo:** Esta tabela está na **1FN**:

Matrícula	Telefones
1111-1	12341234
1111-1	98762334
1111-2	12585897
1111-2	12547895
1111-3	12348792
1111-4	12345874

Segunda Forma Normal

- Uma relação está na **2FN** se está na **1FN** e todo atributo não primário tem dependência funcional total da chave primária da relação;
- O teste é aplicado apenas às relações cuja chave primária é composta por mais de um atributo;
- Quando um atributo não primário puder ser determinado apenas por uma parte da chave primária, a tabela não está na **2FN**.

Segunda Forma Normal

- Vamos supor a seguinte de relação, onde a chave primária é composta pelos atributos Empregado e CodProjeto;

Empregado	CodProjeto	NumHoras
1111-1	2	12
1111-1	3	12
1111-2	1	12
1111-4	2	12
1111-4	3	12

Segunda Forma Normal

- A tabela em questão está na **2FN**:
 - O único atributo não primário, que é NumHoras, só pode ser determinado pela combinação dos dois atributos da chave primária;
 - Caso um deles seja retirado, não podemos determinar com precisão o valor deste atributo.

Segunda Forma Normal

- Vamos supor agora a seguinte de relação, onde a chave primária é composta pelos atributos Empregado e CodProjeto:

Empregado	CodProjeto	NomeEmpregado	NumHoras
1111-1	2	João	12
1111-1	3	João	12
1111-2	1	Maria	12
1111-4	2	Joaquim	12
1111-4	3	Joaquim	12

Segunda Forma Normal

- A tabela em questão não está na **2FN**:
 - O atributo NumHoras não viola a **2FN**, pois só pode ser determinado pelos dois atributos da chave primária;
 - O atributo NomeEmpregado viola a **2FN**, pois pode ser determinado funcionalmente apenas pelo atributo matrícula.

Segunda Forma Normal

- Para normalizar na **2FN** uma tabela que não está no grau 2, devemos criar novas relações para abrigar os atributos que violam a restrição;
- Nas novas tabelas, o valor dos atributos será determinado apenas pela chave parcial da qual é dependente;
- No exemplo anterior, normalizamos a tabela na **2FN** se colocarmos o atributo NomeEmpregado em outra tabela, onde ele dependa funcionalmente apenas do atributo Matrícula.

Terceira Forma Normal

- Uma tabela está na **3FN** se ela está na **2FN** e se nenhum atributo não primário da relação for transitivamente dependente da chave primária:
 - Cada atributo da relação só pode ser funcionalmente dependente da chave primária (totalmente) ou de uma chave candidata da mesma.

Terceira Forma Normal

- Vamos supor a seguinte tabela, onde Matrícula é a chave primária:

Matrícula	Nome	Salário	CodDept o	GerenteDept o
1111-1	João	2500	1	1111-4
1111-2	Maria	2500	2	1111-3
1111-3	Carlos	4500	2	1111-3
1111-4	Joaqui m	4500	1	1111-4

Terceira Forma Normal

- A tabela em questão está na **2FN**, mas não está na **3FN**:
 - Dependência Transitiva:
 - Matrícula \longrightarrow CodDepto \longrightarrow GerenteDepto;
 - Note que o atributo CodDepto determina funcionalmente o atributo GerenteDepto;
 - CodDepto não é chave candidata nem faz parte da chave primária da relação;
 - Esta dependência transitiva viola a **3FN**.

Terceira Forma Normal

- Podemos normalizar no grau 3 uma relação que não está na **3FN** decompondo a relação em mais tabelas de forma que:
 - A tabela contenha o atributo que tem a dependência transitiva;
 - A tabela contenha o atributo não chave que causa a dependência transitiva;
 - O atributo não chave é usado como chave primária da nova tabela.

Terceira Forma Normal

- No nosso exemplo, colocamos a tabela em questão na 3FN criando a seguinte tabela, que tem como chave primária o atributo CodDepto;

<u>CodDepto</u>	GerenteDept o
1	1111-4
2	1111-3

0 grau de normalização

- Para cada tabela, testamos as três formas normais, na seguinte ordem **1FN, 2FN e 3FN**;
- O grau de normalização corresponde ao maior grau de normalização obtido para a relação:
 - Uma relação tem grau de normalização 1 quando está na 1FN mas não está na 2FN;
 - Uma relação tem grau de normalização 2 se está na 1FN mas não está na 3FN;
 - Uma relação tem grau de normalização 3 se está nas três formas normais.

0 grau de normalização

- Um bom projeto de BD visa sempre manter todas as tabelas normalizadas no grau 3;
- Infelizmente, nem sempre isso é possível:
 - Muitas vezes, abre-se mão do grau de normalização para aumentar o desempenho de algumas consultas.



Aula 05 - Dependência Funcional e Normalização



Dúvidas?
vitoria@crateus.ufc.br