

Bases de Dados

PL10 – Normalização de Dados

Docente: Diana Ferreira

Email: diana.ferreira@algoritmi.uminho.pt

Horário de Atendimento:

4ª feira 10h-11h | DI 1.15



Sumário

1 Teoria da Normalização

2 Primeira Forma Normal – 1FN

3 Segunda Forma Normal – 2FN

4 Terceira Forma Normal – 3FN

Bibliografia:

- Connolly, T., Begg, C., Database Systems, A Practical Approach to Design, Implementation, and Management , Addison-Wesley, 4a Edição, 2004. **(Chapter 14 e 15)**

Teoria da Normalização

→ Normalização de Dados

A normalização de dados é uma técnica orientada para a **organização dos dados numa base de dados relacional** que se baseia na análise das chaves primárias e das dependências funcionais de todos os seus atributos.

É um processo **progressivo**, que assenta na execução de uma série de etapas, cada uma delas correspondendo a uma **forma normal** específica com critérios de validação cada vez mais fortes.

Através da sua aplicação, os atributos de um dado modelo de dados são organizados para assegurar a **coesão** dos tipos das entidades envolvidas, minimizando ou mesmo **eliminando duplicação de dados, melhorando a eficiência de armazenamento, a integridade e a escalabilidade** dos dados.

Teoria da Normalização

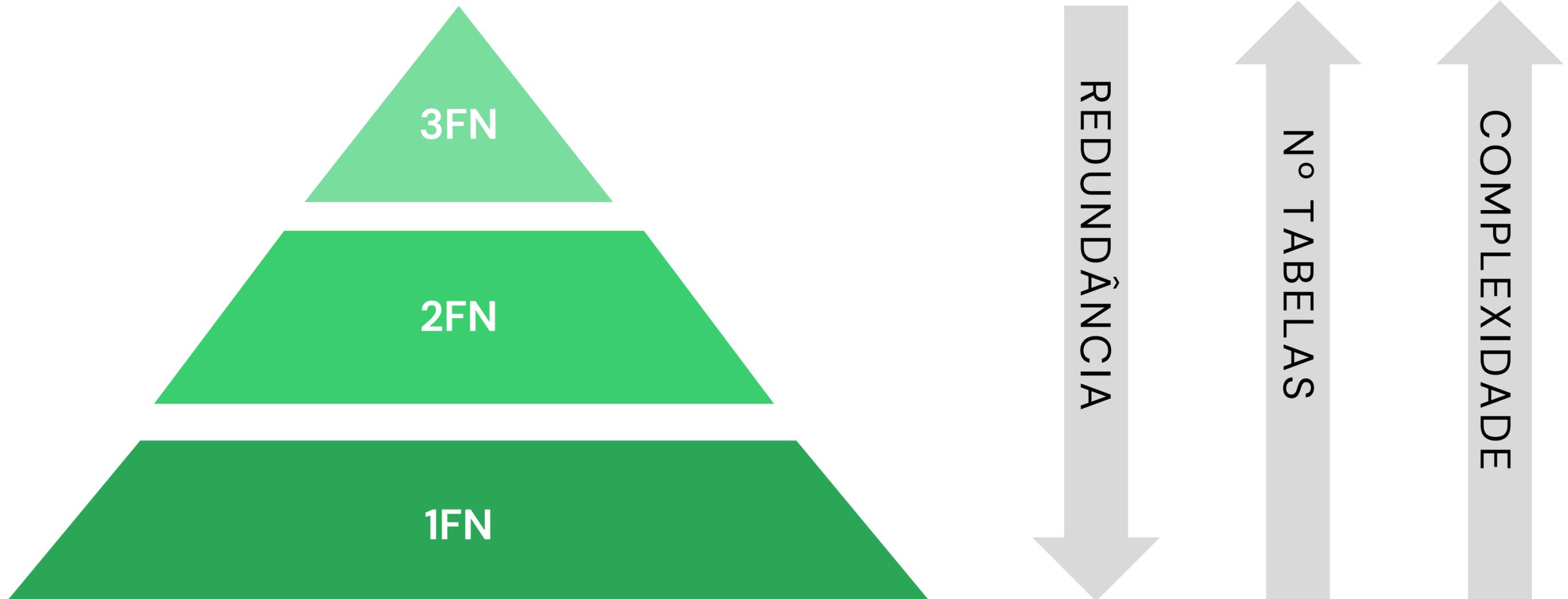
→ Vantagens da Normalização

- Controlo da redundância dos dados;
- Flexibilidade: existem diversas formas de “olharmos” para os dados.
- Garantia de integridade de dados.
- Anomalias de modificações: Na execução de operações de inserção, atualização e remoção.
- Eficiência: Ao se eliminar dados redundantes poupamos algum espaço de armazenamento.

Teoria da Normalização

→ Formas Normais

O processo de normalização é progressivo, ou seja, cada um dos níveis superiores de normalização é um subconjunto do respetivo nível inferior.



Teoria da Normalização

→ Primeira Forma Normal – 1FN

Diz-se que uma relação está na primeira forma normal (1FN) se:

1. Possuir uma chave primária.
2. Todos os seus atributos forem atómicos. Não são permitidos atributos que implicitamente codificam subatributos (atributos compostos) ou atributos multivalor.
3. Não possuir grupos de dados repetidos.

Na prática, podemos dizer que uma relação está na 1FN se as interseções entre colunas (atributos) e linhas (registos) possuírem um único valor – um valor atómico.

Teoria da Normalização

→ Primeira Forma Normal – 1FN

Aplicação da 1ª Forma Normal:

Passo 1: Uma das chaves candidatas é escolhida para chave primária.

Passo 2: Atributos multivalor são convertidos em novas relações com chave externa referindo a chave primária da tabela original.

Passo 3: Cada atributo composto é mapeado em vários sub-atributos atômicos.

Teoria da Normalização

→ Primeira Forma Normal – 1FN

EXEMPLO:

Considere o caso de uma universidade onde os alunos frequentam várias disciplinas. Assim sendo, considerando que os alunos e a respetiva avaliação disciplinar são representados por uma única relação:

Notas(id_aluno, nome_aluno, cod_curso, nome_curso, { disciplina(cod_dis, nome_dis, nota) }, cod_prof, nome_prof, dep_prof)

Notas
id_aluno
nome_aluno
cod_curso
nome_curso
{disciplina(cod_dis, nome_dis, nota)}
cod_prof
nome_prof
dep_prof

Relação não normalizada!



Teoria da Normalização

→ Primeira Forma Normal – 1FN

Notas(id_aluno, nome_aluno, cod_curso, nome_curso, { disciplina(cod_dis, nome_dis, nota) }, cod_prof, nome_prof, dep_prof)

Notas
id_aluno
nome_aluno
cod_curso
nome_curso
{disciplina(cod_dis, nome_dis, nota)}
cod_prof
nome_prof
dep_prof

Atributo composto e multivalor!

(001, 'João Ferreira', 'C01', 'MIEI', 'D01', 'Bases de Dados', 16, 'P01', 'Maria do Carmo', 'Dep. Informática')
 (001, 'João Ferreira', 'C01', 'MIEI', 'D02', 'Criptografia', 12, 'P02', 'Paulo Gomes', 'Dep. Informática')
 (002, 'Rita Abreu', 'C01', 'MIEI', 'D01', 'Bases de Dados', 17, 'P01', 'Maria do Carmo', 'Dep. Informática')
 (002, 'Rita Abreu', 'C01', 'MIEI', 'D03', 'Estatística Aplicada', 14, 'P03', 'Tiago Pinho', 'Dep. Sistemas')

Grupo repetitivo!

Teoria da Normalização

→ Primeira Forma Normal – 1FN



Alunos(id_aluno, nome_aluno,
 cod_curso, nome_curso)

Alunos
<u>id_aluno</u>
nome_aluno
cod_curso
nome_curso

(001, 'João Ferreira', 'CO1', 'MIEI')
 (001, 'João Ferreira', 'CO1', 'MIEI')
 (002, 'Rita Abreu', 'CO1', 'MIEI')
 (002, 'Rita Abreu', 'CO1', 'MIEI')

...

Notas(id_aluno, cod_dis, nome_dis, nota, cod_prof, nome_prof, dep_prof)

Notas
<u>id_aluno</u>
<u>cod_dis</u>
nome_dis
nota
cod_prof
nome_prof
dep_prof

(001, 'D01', 'Bases de Dados', 16, 'PO1', 'Maria do Carmo', 'Dep. Informática')
 (001, 'D02', 'Criptografia', 12, 'PO2', 'Paulo Gomes', 'Dep. Informática')
 (002, 'D01', 'Bases de Dados', 17, 'PO1', 'Maria do Carmo', 'Dep. Informática')
 (002, 'D03', 'Lógica Computacional', 14, 'PO3', 'Tiago Pinho', 'Dep. Sistemas')

...

Teoria da Normalização

→ Segunda Forma Normal – 2FN

Diz-se que uma relação está na segunda forma normal (2FN) se:

1. Lembra-se que, a aplicação das diversas formas normais é progressiva. Assim, para que uma dada relação esteja na 2FN tem de estar também na 1FN.
2. Todos os seus atributos não-primos forem totalmente **dependentes** da sua chave primária. Diz-se que um atributo é não-primo quando este não faz parte de uma chave primária. Ou seja, nenhum atributo pode depender funcionalmente de uma chave parcial.

O que é uma dependência funcional?

As dependências funcionais determinam a forma como se pode interpretar e relacionar os dados e permitem especificar medidas formais sobre a correção dos esquemas relacionais. Estas são definidas com base nas restrições aplicacionais que se verifiquem sobre os atributos no contexto de um dado problema real.

Teoria da Normalização

→ Segunda Forma Normal – 2FN

Na prática, a dependência funcional $A1 \rightarrow A2$ entre dois conjuntos de atributos de uma relação significa que:

- para cada valor de A1, existe apenas um valor possível para A2, por isso diz-se que A2 é funcionalmente dependente de A1 ou que A1 determina funcionalmente A2;
- valores iguais para A1, determinam valores iguais para A2.

Teoria da Normalização

→ Segunda Forma Normal – 2FN

Aplicando a análise de dependências funcionais ao caso de estudo anterior:

Alunos(id_aluno, nome_aluno, cod_curso, nome_curso)

Notas(id_aluno, cod_dis, nome_dis, nota, cod_prof, nome_prof, dep_prof)

Diagrama de Dependências

id_aluno → nome_aluno, cod_curso
cod_curso → nome_curso
{id_aluno, cod_dis} → nota
cod_disc → nome_dis, cod_prof
cod_prof → nome_prof, dep_prof



A tabela Alunos está na 2FN, mas a Notas não!

Teoria da Normalização

→ Segunda Forma Normal – 2FN

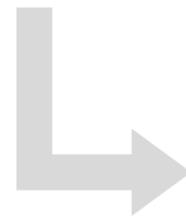
Diagrama de Dependências

{id_aluno, cod_dis} → nota
 cod_disc → nome_dis, cod_prof
 cod_prof → nome_prof, dep_prof

Notas(id_aluno, cod_dis, nome_dis, nota, cod_prof, nome_prof, dep_prof)

<u>id_aluno</u>	<u>cod_dis</u>	nome_dis	nota	cod_prof	nome_prof	dep_prof
001	D01	Bases de Dados	16	P01	Maria do Carmo	Dep. Informática
001	D02	Criptografia	12	P02	Paulo Gomes	Dep. Informática
002	D01	Bases de Dados	17	P01	Maria do Carmo	Dep. Informática
002	D03	Lógica Computacional	14	P03	Tiago Pinho	Dep. Sistemas

... ..



Dependência Parcial

Teoria da Normalização

→ Segunda Forma Normal – 2FN



Notas(id_aluno, cod_dis, nota)

Notas
<u>id_aluno</u>
<u>cod_dis</u>
nota

(001, 'D01', 16)
 (001, 'D02', 12)
 (002, 'D01', 17)
 (002, 'D03', 14)
 ...

Disciplinas(cod_dis, nome_dis, cod_prof, nome_prof, dep_prof)

Disciplinas
<u>cod_dis</u>
nome_dis
cod_prof
nome_prof
dep_prof

('D01', 'Bases de Dados', 'P01', 'Maria do Carmo', 'Dep. Informática')
 ('D02', 'Criptografia', 'PO2', 'Paulo Gomes', 'Dep. Informática')
 ('D01', 'Bases de Dados', 'P01', 'Maria do Carmo', 'Dep. Informática')
 ('D03', 'Lógica Computacional', 'P03', 'Tiago Pinho', 'Dep. Sistemas')
 ...

Teoria da Normalização

→ Terceira Forma Normal – 3FN

Diz-se que uma relação está na terceira forma normal (3FN) se:

1. Lembra-se que, a aplicação das diversas formas normais é progressiva. Assim, para que uma dada relação esteja na 3FN tem de estar também na 1FN e 2FN.
2. Todos os seus atributos que não sejam chaves primárias sejam mutuamente independentes, não havendo assim dependências funcionais transitivas. Por outras palavras, numa relação na 3FN, todos os atributos dependem única e exclusivamente da chave primária.

Na prática, isto significa que os atributos que não dependam da chave primária devem ser “eliminados” da relação, ou seja, devem ser transferidos para outra tabela.

Teoria da Normalização

→ Terceira Forma Normal – 3FN

Na prática, a dependência funcional $A1 \rightarrow A2, A3$ e $A3 \rightarrow A4$ entre os atributos de uma relação significa que:

- Existe uma dependência funcional transitiva entre A1 e A4. Ou seja os atributos que não são chave primária, não são mutuamente independentes entre si.

Aplicando a análise de dependências funcionais ao caso de estudo anterior:

Diagrama de Dependências

Alunos(id_aluno, nome_aluno, cod_curso, nome_curso)

Notas(id_aluno, cod_dis, nota)

Disciplinas(cod_dis, nome_dis, cod_prof, nome_prof, dep_prof)

id_aluno → nome_aluno, cod_curso
 cod_curso → nome_curso
 {id_aluno, cod_dis} → nota
 cod_disc → nome_dis, cod_prof
 cod_prof → nome_prof, dep_prof

A tabela Alunos e Disciplinas não estão na 3FN!

Teoria da Normalização

→ Terceira Forma Normal – 3FN

Diagrama de Dependências

id_aluno → nome_aluno, cod_curso
 cod_curso → nome_curso

Alunos(id_aluno, nome_aluno, cod_curso, nome_curso)

<u>id_aluno</u>	<u>nome_aluno</u>	cod_curso	nome_curso
001	João Ferreira	C01	MIEI
001	João Ferreira	C01	MIEI
002	Rita Abreu	C01	MIEI
002	Rita Abreu	C01	MIEI
...

Dependência Transitiva

Teoria da Normalização

→ Terceira Forma Normal – 3FN

Diagrama de Dependências

cod_disc → nome_dis, cod_prof
cod_prof → nome_prof, dep_prof

Disciplinas(cod_dis, nome_dis, cod_prof, nome_prof, dep_prof)

<u>cod_dis</u>	nome_dis	cod_prof	nome_prof	dep_prof
D01	Bases de Dados	P01	Maria do Carmo	Dep. Informática
D02	Criptografia	P02	Paulo Gomes	Dep. Informática
D01	Bases de Dados	P01	Maria do Carmo	Dep. Informática
D03	Lógica Computacional	P03	Tiago Pinho	Dep. Sistemas

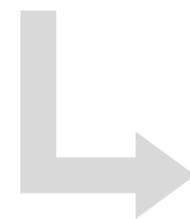
...

...

...

...

...



Dependência Transitiva

Teoria da Normalização

→ Terceira Forma Normal – 3FN



Alunos(id_aluno, nome_aluno, cod_curso)

Alunos
<u>id_aluno</u>
nome_aluno
cod_curso

Cursos(cod_curso, nome_curso)

Cursos
<u>cod_curso</u>
nome_curso



Disciplinas(cod_dis, nome_dis, cod_prof)

Disciplinas
<u>cod_dis</u>
nome_dis
cod_prof

Professores(cod_prof, nome_prof, dep_prof)

Professores
<u>cod_prof</u>
nome_prof
dep_prof

Teoria da Normalização

→ Terceira Forma Normal – 3FN

Alunos(id_aluno, nome_aluno, cod_curso)

Alunos
<u>id_aluno</u>
nome_aluno
cod_curso

Disciplinas(cod_dis, nome_dis, cod_prof)

Disciplinas
<u>cod_dis</u>
nome_dis
cod_prof

Cursos(cod_curso, nome_curso)

Cursos
<u>cod_curso</u>
nome_curso

Professores(cod_prof, nome_prof, dep_prof)

Disciplinas
<u>cod_prof</u>
nome_prof
dep_prof

Diagrama de Dependências

$id_aluno \rightarrow nome_aluno, cod_curso$
 $cod_curso \rightarrow nome_curso$
 $\{id_aluno, cod_dis\} \rightarrow nota$
 $cod_disc \rightarrow nome_dis, cod_prof$
 $cod_prof \rightarrow nome_prof, dep_prof$

As relações encontram-se na 3FN!

Próxima aula: Exploração da BD

- Procedures
- Triggers
- Functions
- Handlers

Exploração

