

# Refinamento de Esquemas e Formas Normais

André Luis Schwerz  
andreluis@utfpr.edu.br

Universidade Tecnológica Federal do Paraná

Banco de Dados 1  
2021/2

# Agenda

- 1 Contextualização
- 2 Dependências Funcionais
- 3 Problemas com a Decomposição
- 4 Normalização
- 5 1a. Forma Normal (1FN)
- 6 2a. Forma Normal (2FN)
- 7 3a. Forma Normal (3FN)
- 8 Exercício Resolvido

- O projeto conceitual possui **esquemas de relações** e **restrições de integridade (RI)**
  - O modelo ER é um bom ponto de partida para o projeto de BD
    - Abordagem Top-Down
  - Esquemas gerados a partir do modelo ER NÃO são perfeitos
  - Problemas causados pela redundância
    - Armazenamento redundante, anomalias de atualização, de inserção, de remoção
  - O projeto inicial deve ser refinado afim de eliminar redundâncias, mas atendendo a critério de desempenho

- As **dependências funcionais** podem ser usadas para identificar esquemas com os problemas e anomalias citadas
- Dependências Funcionais (DF) englobam algumas RIs
  - Um DF ocorre quando um conjunto de atributos é determinado por outro conjunto de atributo, ou ele é dependente de outro conjunto de atributos.
- A partir da detecção das DFs, o refinamento é realizado por meio de decomposições

# Exemplo

- Considere o esquema de relação
  - Funcion\_Horista2 (cpf, nome, vaga, avaliacao, salario\_hora, horas\_trabalhadas)

C	N	V	A	S	H
123-22-3666	Attishoo	48	8	10	40
231-31-5368	Smiley	22	8	10	30
131-24-3650	Smethurst	35	5	7	30
434-26-3751	Guldu	35	5	7	32
612-67-4134	Madayan	35	8	10	40

cpf é a **chave**:  $C \rightarrow CNVASH$

avaliacao **determina** salario\_horas:  $A \rightarrow S$

# Exemplo

- Considere o esquema de relação
  - Alunos (RA, Nome, Nome\_Curso, Idade, Carga\_Horária)

RA	Nome	Nome_Curso	Idade	CH
1	Mario	Computação	21	3200
2	Paulo	Sistemas de Informação	18	3000
3	Almir	Computação	22	3200
4	Marta	Sistemas de Informação	21	3000

RA é a **chave**:  $RA \rightarrow \{Nome, Nome\_Curso, Idade, CH\}$

Nome\_Curso **determina** CH:  $Nome\_Curso \rightarrow CH$

- Uma **DF** (dependência funcional) é um tipo de restrição de integridade que generaliza o conceito de chave
- Dado um esquema de relação  $R$ , com  $X$  e  $Y$  sendo conjuntos não vazios de atributos de  $R$ , dizemos que uma instância  $r$  de  $R$  satisfaz a DF  $X \rightarrow Y$ , se todo par de tuplas  $t_1, t_2$  em  $r$ :
  - Se  $t_1.X = t_2.X$ , então  $t_1.Y = t_2.Y$

- Anomalias
  - **Redundância:** algumas informações são armazenadas repetidamente.
  - **De Atualização:** se uma cópia de dados redundantes é atualizada, pode ocorrer inconsistências se outras cópias são ignoradas
  - **De Inserção:** uma tupla não poderia ser inserida se a dependência não fosse conhecida
  - **De Exclusão:** se as tuplas com um determinado valor fossem excluídas, a associação poderia ser perdida
- Idealmente não se deve permitir redundância
  - Entretanto, há esquemas que permitem a redundância por questões de desempenho



- Como evitar as anomalias?
  - Encontrar as DF
  - Decomposição de um esquema de relação
    - Substituir o esquema por esquemas menores

# Contextualização

## Exemplo de Decomposição

**Tabela:** Funcion\_Horista2 (cpf, nome, vaga, avaliacao, horas\_trabalhadas)

C	N	V	A	H
123-22-3666	Attishoo	48	8	40
231-31-5368	Smiley	22	8	30
131-24-3650	Smethurst	35	5	30
434-26-3751	Guldu	35	5	32
612-67-4134	Madayan	35	8	40

**Tabela:** Salario(avaliacao, salario\_horas)

A	S
8	10
5	7

# Contextualização

## Exemplo de Decomposição

**Tabela:** Aluno2 (RA, Nome, Nome\_Curso, Idade)

RA	Nome	Nome_Curso	Idade
1	Mario	Computação	21
2	Paulo	Sistemas de Informação	18
3	Almir	Computação	22
4	Marta	Sistemas de Informação	21

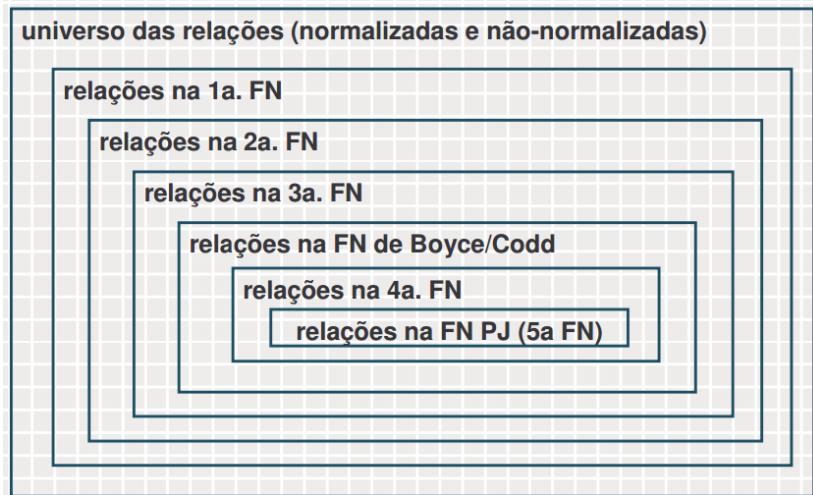
**Tabela:** Curso(Nome\_Curso, Carga\_Horaria)

Nome_Curso	CH
Computação	3200
Sistemas de Informação	3000

# Problemas com a Decomposição

- Decomposição deve ser cautelosa
  - Existe uma razão para decompor a relação?
    - Guias para o refinamento
    - Se um esquema se enquadra em uma determinada forma normal, então sabe-se os tipos de problemas que podem ser evitados/minimizados
  - Quais problemas a decomposição pode causar?
    - Propriedade da **junção sem perda**:  
Recupera-se a instância original a partir das decompostas
    - Propriedade da **preservação da dependência**:  
Restrições impostas nas relações decompostas são impostas também à original
    - Problema do desempenho das consultas

- Processo através do qual esquemas de relação são sucessivamente decompostos até que satisfaçam determinadas propriedades ou formas normais
- As Formas Normais
  - São baseadas nas DFs
  - Garantem consistência das relações
  - Reduzem a redundância e, conseqüentemente, impedem as anomalias



- Uma tabela está na 1FN se, e somente se, todas as colunas tiverem apenas valores atômicos, ou seja, se cada coluna só puder ter um valor para cada linha na tabela.
  - Não pode conter atributo composto
  - Não pode conter atributo multi-valorado
  - Não pode conter conjuntos de atributos repetidos descrevendo a mesma característica
  - Não permite-se relações aninhadas (uma tabela dentro da outra)

# Primeira Forma Normal

- Exemplo de relação que NÃO está na 1FN
- Suponha a relação:
  - PessoaCurso(Nome, Cidade, ID, (Curso)\*)
  - O asterisco (\*) significa que cada Pessoa terá um grupo de Cursos

Nome	Cidade	ID	Cursos
Artur	São Paulo	999	Programador
Ana	Londrina	777	Operador, programador
Carlos	Araruna	888	Analista, programador, operador
Paulo	Maringá	555	Operador, analista

- O atributo Cursos contém valores **não atômicos**



# Primeira Forma Normal

- Mais um exemplo de relação que NÃO está na 1FN

Nome	Cidade	ID	Curso 1	Curso 2	Curso 3
Artur	São Paulo	999	Programador		
Ana	Londrina	777	Operador	Programador	
Carlos	Araruna	888	Analista	Programador	Operador
Paulo	Maringá	555	Operador	Analista	

- Porque há atributos repetidos do mesmo tipo: Curso 1, Curso 2 e Curso 3.
- As tuplas correspondentes à alunos com apenas um ou dois cursos terão valores nulos para alguns atributos.
- Como representar uma pessoa com mais do que três cursos?

# Primeira Forma Normal

## Decomposição

- Pessoa (Nome, Cidade, ID)
- PessoaCurso(ID (FK), Curso)

Nome	Cidade	ID
Artur	São Paulo	999
Ana	Londrina	777
Carlos	Araruna	888
Paulo	Maringá	555

ID	Curso
999	Programador
777	Operador
777	Programador
888	Analista
888	Programador
888	Operador
555	Operador
555	Analista

# Primeira Forma Normal

- Mais um exemplo de relação que NÃO está na 1FN

<u>ID</u>	Nome	Telefone	Endereço
1	José	9838-0021 9838-0021	Avenida Brasil, 2033 - São Paulo/SP
2	Maria	8837-0012 2234-1121	Avenida Maranhão, 2933 - Rio de Janeiro/RJ
3	João	9739-0023	Rua Mato Grosso, 44 Belo Horizonte/MG
4	Carlos	3334-1022	Avenida Paulista, 1000 - São Paulo/SP

- Atributo composto (Endereço)
- Atributo multi-valorado (Telefone)

# Primeira Forma Normal

## Decomposição

<u>ID</u>	Nome	Telefone	Logradouro	Cidade	UF
1	José	9838-0021 9838-1221	Avenida Brasil, 2033	São Paulo	SP
2	Maria	8837-0012 2234-1121	Avenida Maranhão, 2933	Rio de Janeiro	RJ
3	João	9739-0023	Rua Mato Grosso, 44	Belo Horizonte	MG
4	Carlos	3334-1022	Avenida Paulista, 1000	São Paulo	SP

- Atributo composto é decomposto.
- Caberia ainda decompor Logradouro em tipo de logradouro (Rua, Avenida, Travessa), nome do logradouro e número.

# Primeira Forma Normal

## Decomposição

<u>ID</u>	Nome	Logradouro	Cidade	UF
1	José	Avenida Brasil, 2033	São Paulo	SP
2	Maria	Avenida Maranhão, 2933	Rio de Janeiro	RJ
3	João	Rua Mato Grosso, 44	Belo Horizonte	MG
4	Carlos	Avenida Paulista, 1000	São Paulo	SP

<u>ID (FK)</u>	<u>Telefone</u>
1	9838-0021
1	9838-1221
2	8837-0012
2	2234-1121
3	9739-0023
4	3334-1022

- Atributo multi-valorado é decomposto.
- Agora a relação está na Primeira Forma Normal

- Uma tabela está na 2FN se, e somente se,
  - ela estiver na 1FN, e
  - os atributos não-chaves forem totalmente dependentes da chave primária.
- Um atributo será totalmente dependente da chave primária se estiver no lado direito de um DF que tem no lado esquerdo a própria chave primária ou algo que possa ser derivado da chave primária usando a transitividade das DFs.
- Quando um atributo não é totalmente dependente da chave primária dizemos que há um dependência parcial.

# Segunda Forma Normal

- Exemplo de relação que NÃO está na 2FN
  - Pedido(NomeFornecedor, CodPeça, Cidade, Quantidade)
  - Temos que:
    - $\{\text{NomeFornecedor}, \text{CodPeça}\}$  compõem a chave primária
    - $\{\text{NomeFornecedor}, \text{CodPeça}\} \rightarrow \{\text{Quantidade}\}$
    - $\{\text{NomeFornecedor}, \text{CodPeça}\}$  definem Quantidade, mas não definem Cidade.
    - $\{\text{NomeFornecedor}\} \rightarrow \{\text{Cidade}\}$
    - Há, então, uma **dependência parcial**

NomeFornecedor	CodPeça	Cidade	Quantidade
Empresa A	1	Maringá	100
Empresa A	2	Maringá	200
Empresa A	3	Maringá	300
Empresa B	1	Londrina	400
Empresa B	3	Londrina	500

# Segunda Forma Normal

- Algumas anomalias

- Inserção:** Não é possível inserir um fornecedor sem que ele forneça alguma peça (peça faz parte da chave).
- Eliminação:** Se, por exemplo, a empresa B deixar de fornecer as peças 1 e 3, a informação sobre a cidade desse fornecedor será perdida.
- Modificação:** Supondo que um fornecedor muda de cidade. Atualizar a relação, significa atualizar todas as linhas desse fornecedor.

NomeFornecedor	CodPeça	Cidade	Quantidade
Empresa A	1	Maringá	100
Empresa A	2	Maringá	200
Empresa A	3	Maringá	300
Empresa B	1	Londrina	400
Empresa B	3	Londrina	500



# Segunda Forma Normal

## Decomposição

- Pedido(NomeFornecedor (FK), CodPeça, Quantidade)
- Fornecedor(NomeFornecedor, Cidade)

NomeFornecedor	CodPeça	Quantidade
Empresa A	1	100
Empresa A	2	200
Empresa A	3	300
Empresa B	1	400
Empresa B	3	500

NomeFornecedor	Cidade
Empresa A	Maringá
Empresa B	Londrina

# Segunda Forma Normal

- Mais um exemplo de relação que NÃO está na 2FN
  - $\{\text{id-curso}\} \rightarrow \{\text{descrição-curso}\}$

<u>id-aluno</u>	<u>id-curso</u>	nota	descrição-curso
1	1	5.9	Banco de dados
1	2	8.1	Engenharia de software
2	2	9.3	Engenharia de software
3	1	4.6	Banco de dados

# Segunda Forma Normal

## Decomposição

- Agora, ambas estão na Segunda Forma Normal

<u>id-aluno</u>	<u>id-curso</u>	nota
1	1	5.9
1	2	8.1
2	2	9.3
3	1	4.6

<u>id-curso</u>	descrição-curso
1	Banco de dados
2	Engenharia de software

# Segunda Forma Normal

## Resumo

- Assumindo que relação está na 1FN, para vermos se uma relação está na 2FN
  - Identificamos a chave da tabela. Se a chave for apenas um atributo, ou for constituída por todos os atributos da relação, então podemos concluir que está na 2FN.
  - Se a chave for composta (tiver mais do que um atributo) verificamos se há atributos que não são chave e que dependem apenas de parte da chave. Se não houver, então está na 2FN
- Caso contrário, temos que decompor.

# Terceira Forma Normal

- Uma tabela está na 3FN, se, e somente se,
  - Esta na 2FN
  - Nenhum dos atributos não chave depende de outro também não chave (dependência transitiva)

# Terceira Forma Normal

- Exemplo de relação que NÃO está na 3FN
- Analise a relação, Nota(NumNota, CodCliente, NomeCliente, CidadeCliente) em que:
  - {NumNota} é a chave primária
  - {NumNota}  $\rightarrow$  {CodCliente}
- Entretanto,
  - {CodCliente}  $\rightarrow$  {NomeCliente, CidadeCliente}
- E transitivamente:
  - {NumNota}  $\rightarrow$  {NomeCliente, CidadeCliente}

# Terceira Forma Normal

## Decomposição

- Cliente(CodCliente, NomeCliente CidadeCliente)
- Nota(NumNota, CodCliente (FK))

CodCliente	NomeCliente	CidadeCliente
1	Maria	Curitiba
2	Ana	Londrina
3	João	Maringá

NumNota	CodCliente
1	1
2	2
3	2
4	3
5	3

# Execício Resolvido

Indique em que forma normal a relação de se encontra e normalize para a 3FN

ALUNO(RA, NomeA, Status, CodCurso, NomeCurso,  
{DISCIPLINA(CodD, NomeD, CodProf, NomeProf, Nota)})

RA	NomeA	Status	CodCurso	NomeCurso	Disciplina				
					CodD	NomeD	CodProf	NomeProf	Nota
1	Maria	Cursando	1C0	BCC	BCC33E	BD1	1	Andre	10
					BCC33D	SO	2	Rodrigo	5
2	João	Cursando	1C1	SI	CodD	NomeD	CodProf	NomeProf	Nota
					BCC33E	BD1	1	Andre	8
					BCC34E	BD2	1	Andre	6



# Execício Resolvido

## 1o. Passo

ALUNO(RA, NomeA, Status, CodCurso, NomeCurso,  
{DISCIPLINA(CodD, NomeD, CodProf, NomeProf, Nota)})

RA	NomeA	Status	CodCurso	NomeCurso	Disciplina				
					CodD	NomeD	CodProf	NomeProf	Nota
1	Maria	Cursando	1C0	BCC	BCC33E	BD1	1	Andre	10
					BCC33D	SO	2	Rodrigo	5
2	João	Cursando	1C1	SI	CodD	NomeD	CodProf	NomeProf	Nota
					BCC33E	BD1	1	Andre	8
					BCC34E	BD2	1	Andre	6

Não encontra-se na 1FN, pois possui tabelas aninhadas.

# Execício Resolvido

## 1o. Passo - Decomposição

ALUNO(RA, NomeA, Status, CodCurso, NomeCurso)

ALUNO\_DISCIPLINA(RA (FK), CodD, NomeD, CodProf, NomeProf, Nota)

RA	NomeA	Status	CodCurso	NomeCurso
1	Maria	Cursando	1C0	BCC
2	João	Cursando	1C1	SI

RA	CodD	NomeD	CodProf	NomeProf	Nota
1	BCC33E	BD1	1	Andre	10
1	BCC33D	SO	2	Rodrigo	5
2	BCC33E	BD1	1	Andre	8
2	BCC34E	BD2	1	Andre	6

Ambas relações encontram-se na 1FN.

# Execício Resolvido

## 2o. Passo

ALUNO (RA, NomeA, Status, CodCurso, NomeCurso)

ALUNO\_DISCIPLINA (RA (FK), CodD, NomeD, CodProf, NomeProf, Nota)

- $\{RA\} \rightarrow \{NomeA, Status, CodCurso\}$
- $\{CodCurso\} \rightarrow \{NomeCurso\}$
- $\{RA, CodD\} \rightarrow \{Nota\}$
- $\{CodD\} \rightarrow \{NomeD, CodProf\}$
- $\{CodProf\} \rightarrow \{NomeProf\}$

ALUNO está na 2FN, pois não possui dependência parcial.

ALUNO\_DISCIPLINA não está na 2FN, pois  $\{NomeD, CodProf\}$  dependem parcialmente da chave primária.

# Execício Resolvido

## 2o. Passo - Decomposição

ALUNO (RA, NomeA, Status, CodCurso, NomeCurso)

ALUNO\_DISCIPLINA (RA (FK), CodD (FK), Nota)

DISCIPLINA (CodD, NomeD, CodProf, NomeProf)

- $\{RA\} \rightarrow \{NomeA, Status, CodCurso\}$
- $\{CodCurso\} \rightarrow \{NomeCurso\}$
- $\{RA, CodD\} \rightarrow \{Nota\}$
- $\{CodD\} \rightarrow \{NomeD, CodProf\}$
- $\{CodProf\} \rightarrow \{NomeProf\}$

As três relações encontram-se, agora, na 2FN, pois as dependências parciais foram eliminadas.

# Execício Resolvido

## 3o. Passo

ALUNO (RA, NomeA, Status, CodCurso, NomeCurso)

ALUNO\_DISCIPLINA (RA (FK), CodD (FK), Nota)

DISCIPLINA (CodD, NomeD, CodProf, NomeProf)

- $\{RA\} \rightarrow \{NomeA, Status, CodCurso\}$
- $\{CodCurso\} \rightarrow \{NomeCurso\}$
- $\{RA, CodD\} \rightarrow \{Nota\}$
- $\{CodD\} \rightarrow \{NomeD, CodProf\}$
- $\{CodProf\} \rightarrow \{NomeProf\}$

ALUNO e DISCIPLINA possuem dependências transitivas  
ALUNO\_DISCIPLINA já se encontra na 3FN

# Execício Resolvido

## 3o. Passo - Decomposição

ALUNO (RA, NomeA, Status, CodCurso (FK))

CURSO (CodCurso, NomeCurso)

ALUNO\_DISCIPLINA (RA (FK), CodD (FK), Nota)

DISCIPLINA (CodD, NomeD, CodProf (FK))

PROFESSOR (CodProf, NomeProf)

- $\{RA\} \rightarrow \{NomeA, Status, CodCurso\}$
- $\{CodCurso\} \rightarrow \{NomeCurso\}$
- $\{RA, CodD\} \rightarrow \{Nota\}$
- $\{CodD\} \rightarrow \{NomeD, CodProf\}$
- $\{CodProf\} \rightarrow \{NomeProf\}$

As cinco relações encontram-se na 3FN