

# Normalização Avançada

## Conjunto minimo de dependencias funcionais

Um conjunto de dependencias funcionais X é minimo se satisfaz as seguintes condições:

- Todas as dependencias em X têm um unico atributo no seu lado direito.
- Não é possivel substituir qualquer dependencia  $A \rightarrow B$  em X por uma dependencia  $C \rightarrow B$ , onde C é um subset de A, e mesmo assim continuar com um conjunto de dependencias equivalente a X.
- Não é possivel remover alguma dependencia de X e ainda possui um conjunto de dependencias equivalente a X.

## BCFN (Boyce-Codd Normal Form)

- Uma relação está em BCFN se todos os determinantes são chaves candidatas (nem todos os determinantes são chaves).

Considerar o exemplo:

- Fundos consistem de 1 ou mais Tipos de Investimento.
- Fundos são geridos por 1 ou mais Gestores.
- Tipos de Investimento podem ter 1 ou mais Gestores.
- Os Gestores apenas têm 1 tipos de investimentos.

- Relation: FUNDS (FundID, InvestmentType, Manager)

FundID	InvestmentType	Manager
99	Common Stock	Smith
99	Municipal Bonds	Jones
33	Common Stock	Green
22	Growth Stocks	Brown
11	Common Stock	Smith

FD1: FundID, InvestmentType  $\rightarrow$  Manager  
FD2: FundID, Manager  $\rightarrow$  InvestmentType  
FD3: Manager  $\rightarrow$  InvestmentType

Neste caso, a combinação FundID e InvestmentType forma uma chave candidata porque podemos usar FundID, InvestmentType para identificar unicamente cada tuplo numa relação. Similarmente, a combinação FundID e Manager podem também formar uma chave candidata por podemos usar FundID, Manager para identificar unicamente um tuplo.

O Manager por si só não é uma chave candidata porque não podemos usar apenas Manager para identificar unicamente um tuplo na relação.

Esta relação está em 1FN, 2FN e 3FN? Sim. Se escolhermos FundID, InvestmentType como chave primária, está em 1FN.

Está em 2FN porque todos os atributos não chave estão dependentes da chave primária.

Está em 3FN porque não existem dependências transitivas.

No entanto, o que acontece se apagarmos o tuplo com FundID 22? Perdemos o facto de que Brown gere o InvestmentType "Growth Stocks".

Assim, não está em BCFN porque nem todos os determinantes são chaves candidatas. Para efetuar a normalização devemos:

- Fazer a lista de todos os determinantes.
- Ver se cada determinante é uma chave candidata.
- Para cada determinante que não seja chave candidata, criar uma nova relação a partir da dependência funcional e manter o determinante na relação original.

#### 4FN "Forma dos NULLs"

Uma relação está na 4FN se está em BCFN e não possui dependências multivalor.

Dependência multivalor: Um tipo de dependência funcional na qual o determinante pode determinar mais que um valor.

Mais propriamente, possui 3 critérios:

- Tem de haver pelo menos 3 atributos na relação (A, B e C).
- Dado A, este pode determinar múltiplos valores de B.
- Dado A, este pode determinar múltiplos valores de C.
- B e C são independentes entre si.

Exemplo:

- Um estudante tem um ou mais Majors.
- Um estudante tem uma ou mais Atividades.

StudentID	Major	Activities
100	CIS	Baseball
100	CIS	Volleyball
100	Accounting	Baseball
100	Accounting	Volleyball
200	Marketing	Swimming

FD1: StudentID → Major

FD2: StudentID → Activities

Portfolio ID	Stock Fund	Bond Fund
999	Janus Fund	Municipal Bonds
999	Janus Fund	Dreyfus Short-Intermediate Municipal Bond Fund
999	Scudder Global Fund	Municipal Bonds
999	Scudder Global Fund	Dreyfus Short-Intermediate Municipal Bond Fund
888	Kaufmann Fund	T. Rowe Price Emerging Markets Bond Fund

### Características:

- Não existem dependências funcionais regulares.
- Os 3 atributos juntos formam a chave primária.
- Os 2 atributos mais à direita são independentes um do outro.
- Anomalia de inserção: Não é possível inserir uma atividade sem inserir também uma major (ou valor NULL).

Então Atividade e Major formam uma dependência multivalor em StudentID.

StudentID → Major

StudentID → Atividade

Solução: Dividir em duas relações com a chave em comum (StudentID)

### 5FN "Forma dos carros"

A 5FN trata dos casos em que a informação pode ser reconstruída de pedaços mais pequenos de informação que possuem menos redundância (2FN, 3FN e 4FN também trata disto mais 5FN refere-se a casos não abordados pelos outros).

### Exemplo:

Se Agents representam Companies, Companies fabricam Products e Agents vendem Products, poderemos querer guardar a informação relativa a que Agent vende que Produto de que Company. A informação poderia ser guardada numa única relação:

AGENT	COMPANY	PRODUCT
Smith	Ford	car
Smith	GM	truck

Embora o Agent Smith venda carros fabricados pela Ford e trucks fabricados pela GM, ele não vende trucks fabricados pela Ford ou carros fabricados pela GM. Então necessitamos da combinação dos 3 campos para saber que combinações são ou não válidas.

Mas suponhamos que uma certa regra estava em efeito: se um Agente vende um certo Product, e ele representa a Company que fabrica esse Product, então ele vende esse Product para esse Company.

AGENT	COMPANY	PRODUCT
Smith	Ford	car
Smith	Ford	truck
Smith	GM	car
Smith	GM	truck
Jones	Ford	car

Neste caso, podemos reconstruir todos os factos a partir de uma forma normalizada que consiste em 3 relações separadas:

AGENT	COMPANY
Smith	Ford
Smith	GM
Jones	Ford

COMPANY	PRODUCT
Ford	car
Ford	truck
GM	car
GM	truck

AGENT	PRODUCT
Smith	car
Smith	truck
Jones	car

Estas 3 relações estão em 5FN, enquanto que a tabela acima (a que tem 5 entradas) não está.

De forma lata, podemos dizer que uma relação se encontra em 5FN se a sua informação não pode ser reconstruída de pedaços de relações mais pequenas, e então, de relações que possuam menos atributos que a relação original. Se uma relação pode ser decomposta em relações mais pequenas que possuem todas a mesma chave primária, então considera-se que essa relação se encontra em 5FN sem decomposição. Uma relação em 5FN também se encontra em todas as outras FN's.

A 5FN não difere da 4FN a não ser que exista uma restrição simétrica tal como a regra sobre os Agents, Companies e Products. Na ausência de essa restrição, uma relação em 4FN também se encontra em 5FN.

Uma vantagem da 5FN é que certas redundâncias podem ser eliminadas. Na forma normalizada, o facto de que o Smith vende carros é apenas guardado uma vez. Na forma desnormalizada, pode ter de ser guardado várias vezes.

Embora a forma normalizada envolva mais relações, poderá haver um número menor de tuplos. A vantagem é notável à medida que o tamanho das relações cresce, já que estas crescem em forma aditiva, enquanto que na forma desnormalizada crescem de forma multiplicativa. Por exemplo, se adicionamos um novo Agent que vende  $x$  Products e trabalha para  $y$  Companys, em que cada uma destas Companys faz cada um destes Products, temos de adicionar  $x+y$  entradas na forma normalizada, mas  $x*y$  entradas na desnormalizada.