Formas Normais

Índice

[1. Formas Normais de BD relacionais 3](#_Toc473113040)

[1.1 Primeira forma normal (1FN ou 1NF). Atributos multivaluados ou repetitivos 3](#_Toc473113041)

[Pasos para converter unha relación en 1FN: 3](#_Toc473113042)

[Pasos: 4](#_Toc473113043)

[1.2 Segunda forma normal (2FN ou 2NF). Dependencia de clave 5](#_Toc473113044)

[Proceso de descomposición formal: 6](#_Toc473113045)

[Pasos para converter relacións en 1FN a 2FN 7](#_Toc473113046)

[1.3 Terceira forma normal (3FN ou 3NF). Dependencias transitivas 8](#_Toc473113047)

[Proceso de descomposición formal: 8](#_Toc473113048)

[Pasos 9](#_Toc473113049)

[1.4 Forma normal de Boyce-Codd (FNBC ou BCNF). Dependencia de la clave 11](#_Toc473113050)

1. Formas Normais de BD relacionais
   1. Primeira forma normal (1FN ou 1NF). Atributos multivaluados ou repetitivos

Un esquema R cumpre a primeira forma normal (1FN) se os dominios de todos os atributos son atómicos e univaluados, ningún atributo pode tomar valores compostos (dominio non atómico) nin multivaluados (mais dun valor ao mesmo tempo), estando libre de grupos repetitivos (táboas dentro de táboas). Esta forma normal é parte inherente do modelo relacional de datos.

Un atributo é atómico se os seus elementos pódense considerar como unidades indivisibles, exemplos de dominios non átomicos son:

* Nome completo: atributo composto polo nome máis os apelidos.
* Teléfonos: atributos multivaluado, xa que por regra xeral os contactos soen posuir máis dun número de teléfono (fixo, móbil, persoal, traballo, etc).

**Moitas veces, a definición do que é un valor atómico non é clara, e pode depender dos usos que se vaian facer dos valores do atributo. Por exemplo, cando se trata de procesar datos de persoas pode ter sentido utilizar un atributo “nome completo” para almacear o nome propio e os apelidos, se está información sempre se abordar de forma conxunta. En cambio, si se desexa acceder polo nome propio e/ou polos apelidos separadamente, para que o esquema de relación estea en 1FN debería ter atributos diferenciados para eles, “nomePropio”, “apelido1” “apelido2”**.

Pasos para converter unha relación en 1FN:

Cando unha relación non está en primeira forma normal, divídese noutras relacións, re-partindo os seus atributos entre as resultantes. Normalmente a idea é eliminar o atributo que incumpre a 1ª FN da relación orixinal e colocalo nunha relación diferente xunto coa clave primaría da relación de partida:

No exemplo o punto de partida é o seguinte formulario:

Formulario.emf

Figura. Formulario dun pedido

Pódese intuír a seguinte relación non normalizada que conten elementos repetidos e subtaboas (características non permitidas no modelo relacional). No esquema de relación márcase como clave o atributos codPedido, que cumpre a condición de ser mínima e identifica univocamente cada unha das tuplas sen repetirse:

*PEDIDO (codPedido,(codProducto, descrición, prezoUnidade, cantidade, total), data, idCliente, nomeCliente, cidade, telefono, totalPedido)*

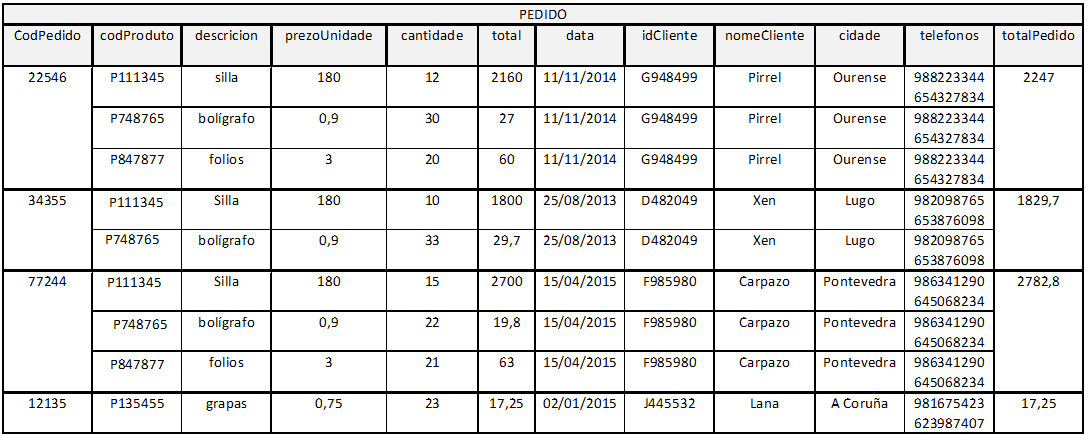


Figura. Relación PEDIDO non normalizada extraída do formulario PEDIDO

Pasos:

Todos os atributos deben ser atómicos e univaluados, ningún pode tomar valores compostos (dominio non atómico) nin multivaluados (mais dun valor ao mesmo tempo), estando libre de grupos repetitivos (táboas dentro de táboas).

* I) Crear unha nova relación cos atributos que incumpren ser atómico ou son táboas dentro de táboas. No exemplo comenzaremos coa subtáboa que se atopa dentro de cada tupla que representa un pedido completo:

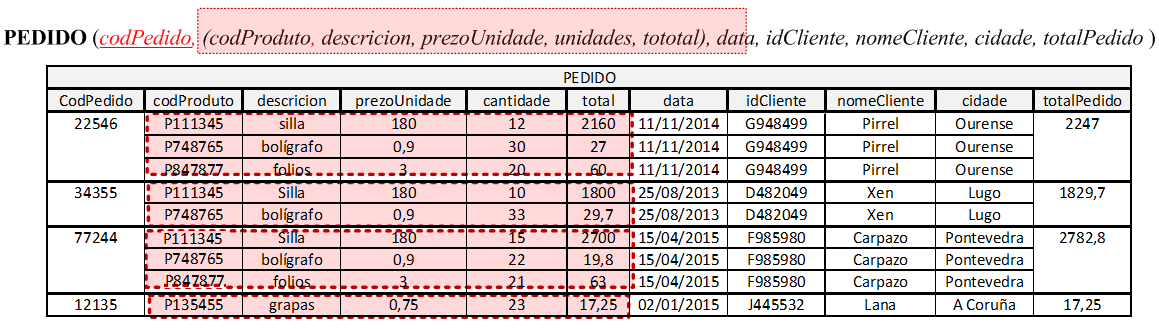


Figura. Relación PEDIDO non normalizada que contén grupos repetitivos

* II) Engadir a esta nova relación a clave primaria da relación que orixinalmente a contiña.
* III) Darlle un nome á nova relación (opcionalmente no nome incluirase a FN na que a táboa aparece).

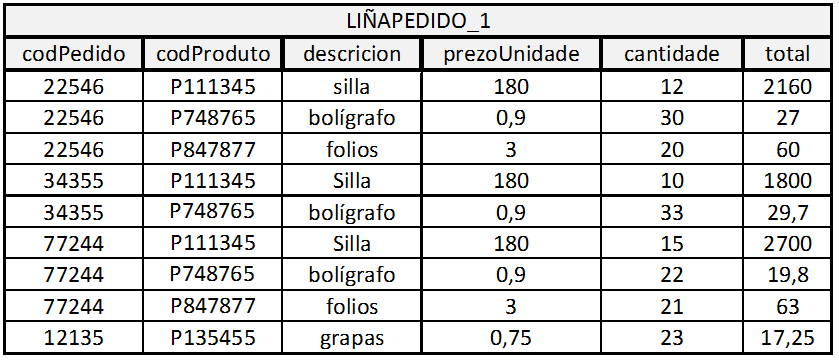


Figura. Relación resultante do proceso de normalización a 1FN ao eliminar os elementos repetitivos

* IV) Determinar a clave primaria da nova relación: serán os atributos que identifican univocamente cada tupla da táboa sendo mínima. No exemplo a clave resultante é composta por CodPedido e CodProduto.

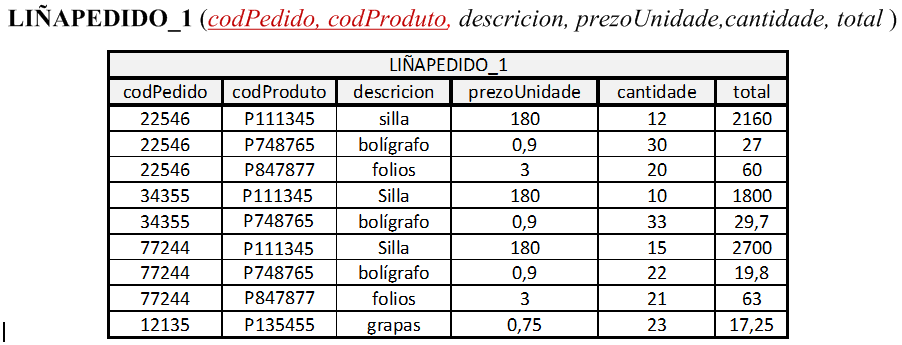


Figura. Relación resultante tralo proceso de detección de clave

* V) Repetir ata que non queden máis atributos non atómicos, no exemplo repitese o proceso para o tributo multivaluado télefonos.
  1. Segunda forma normal (2FN ou 2NF). Dependencia de clave

Unha relación R está en 2FN, se está 1FN e todo atributo que non sexa clave primaria, nin forme parte dela, ten dependencia funcional completa ou plena respecto da clave primaria desa relación, é dicir, o valor dos atributos non principais da relación ven determinado polo valor de todos os atributos da clave. Só os atributos dunha clave poden depender de partes dunha clave.

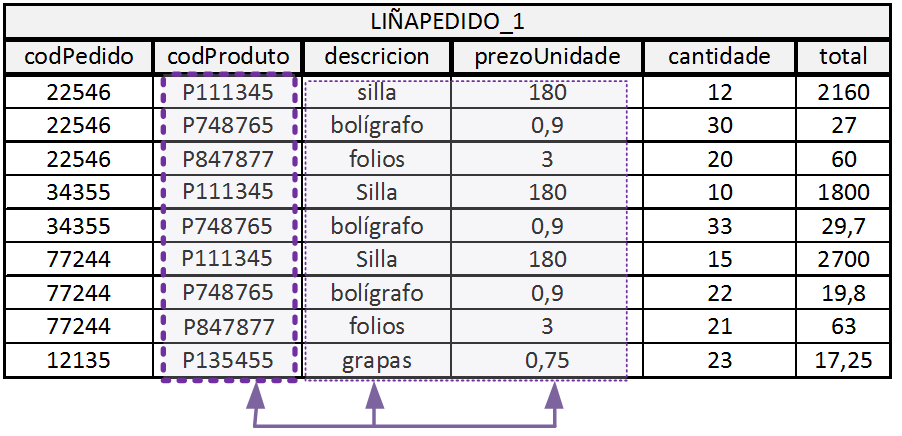


Figura. Relación en 1FN con atributos (descrición e prezoUnidade) que dependen de parte da clave codProduto

No exemplo observase como certos atributos só dependen de parte da clave, coma descrición e prezoUnidade que dependen funcionalmente do atributo codProduto pero non de codPedido polo que as dependencias funcionais (df) son:

* codProduto -> descricion.
* codProduto -> prezoUnidade.

Esta forma normal só se aplica a relacións que teñen claves compostas, é dicir, que están formadas por máis dun atributo. Se un esquema de relación non está en 2FN se normaliza en varias relacións que si estean en 2FN, nas que os atributos que dependen dunha parte da clave formarán unha nova relación que terá esa parte da clave como clave primaria.

Proceso de descomposición formal:

Formalmente pódese definir o proceso de descomposición da seguinte maneira: Dada unha relación R con clave composta da forma (R.x, R.y), e en esta relación existe una dependencia funcional incompleta de la forma R.y ->R.z, onde R.z é un atributo que non pertence á clave) da relación R, o proceso de descomposición realizarase da forma seguinte:

* Da relación R elimínase o atributo R.z, quedando igual o resto.
* Constrúese unha nova relación R1, coa estrutura:
* O atributo R.z como atributo non clave da relación R1.
* O atributo R.y como atributo clave (clave primaria) da relación R1.
* En R defínese a R.y como clave foránea da relación R1, (R1.y).

A descomposición para a aplicación da 2FN débese facer en base á dependencia funcional incompleta, e nunca por outras dependencia entre atributos da relación. Neste proceso tanto R.x, R.y, R.z poden ser a súa vez atributos simples ou agregados de datos.

Pasos para converter relacións en 1FN a 2FN

Normalizarase o esquema do exemplo seguindo os seguintes pasos:

* I) Eliminar os atributos que dependen parcialmente da clave primaria e crea con eles unha nova relación (sen elementos repetidos).
* II) Nomear á nova relación (engadir opcionalmente un 2 para indicar 2NF).
* III) Engadir a esta relación unha copia do atributo ou atributos do cales dependen (será á clave primaria da nova relación).

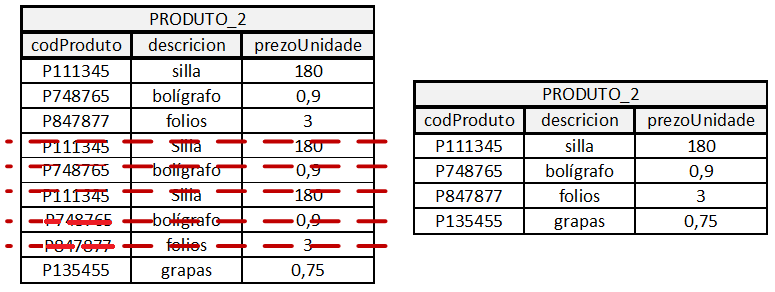
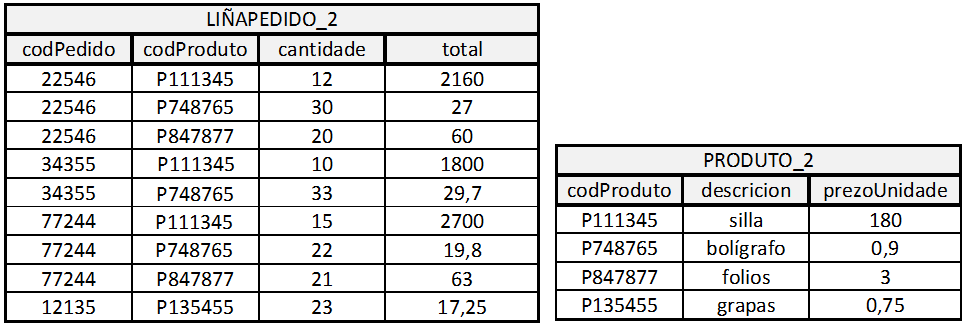


Figura. Táboa resultante do proceso de normalización a 2FN

* IV) Renomear á relación orixinal (engadir un 2 para indicar 2FN)



subModelo2FN.emf

Figura. Táboas é grafo relacional obtido tras normalizar en 2FN

* V) Repetir o proceso ata que todas as táboas adopten a 2FN

A táboa orixinaria, o eliminar de ela os elementos repetitivos tamén se atopa en 2FN:

Pedido_2fn.emf

Figura. Táboa en 2FN tra a normalización

* 1. Terceira forma normal (3FN ou 3NF). Dependencias transitivas

Unha relación está en terceira forma normal se cumpre as seguintes condicións:

* A relación está en 2FN.
* Todos os atributos da relación dependen funcionalmente só da clave, e non de ningún outro atributo non clave (todos os atributos non principais son independentes)
* Non existen dependencias funcionais transitivas respecto das claves. Un atributo depende transitivamente da clave primaria si depende doutro atributo que depende da clave

Resumindo, para toda DF:X ->Y, X é unha clave.

Proceso de descomposición formal:

Dada unha relación R de clave R.x e dous atributos non clave R.y e R.z, e nesta relación están presentes as seguintes dependencias funcionais:

R.x->R.y, R.y -> R.z e, por tanto a dependencia transitiva, R.x -> R.z, o proceso de descomposición realizarase da forma seguinte:

* Da relación R elimínase o atributo que mantén unha dependencia funcional transitiva coa clave da relación, é dicir, o atributo R.z, quedando igual ao resto da relación R.
* Constrúese unha nova relación R1, coa estrutura seguinte:
* O atributo R.z que mantiña unha dependencia funcional transitiva.
* O atributo R.y co cal o atributo R.z mantiña unha dependencia funcional completa.
* A clave da relación R1, será o atributo R1.y e, nesta relación só existirá unha dependencia funcional completa da forma: R1.y -> R1.z.

Na relación R defínese R.y como clave allea da relación R1 (R1.y).

Deste xeito, a descomposición que se debe realizar por aplicación da 3FN debe ser en base á dependencia funcional transitiva e nunca por outra dependencia entre os atributos da relación. Neste proceso tanto R.x, R.y, R.z poden ser a súa vez atributos simples ou agregados de datos.

Pasos

No exemplo, a táboa PEDIDO\_2 presenta anomalías por dependencias funcionais transitivas, seguimos os seguintes pasos para que adopte a 3FN:

* I) Comprobar que cada relación ten un número fixo de columnas e as variables son sinxelas (atómicas).
* II) Identifica a clave primaria.
* III) Comproba que todos os atributos (menos a clave primaria) depende de TODA a clave non de PARTE dela.
* IV) Si existe dependencia parcial dividir a relación en varias subrelacións.
* V) Comprobar que todos os atributos dependen da clave e non doutros atributos (o que se coñece coma dependencias transitivas): Observase como os atributos nomeCliente, cidade non dependen da clave ou de parte de ela senón do atributo idCliente que a súa vez si depende de codPedido, producese unha dependencia transitiva:

Producto2FN_Dependencias transitivas.emf

Figura. Táboa con dependencias transitivas de CodPedido a idCliente e de idCliente a nomeCliente e cidade

* VI) Si existe dependencias non relacionadas con clave primaria dividir as relación.

ClientePedido3Fn.emf

Figura . Táboas e esquema obtidas tras a normalización a 3FN

A maioría das táboas en 3NF están libres de anomalías de actualización, inserción, e borrado. Certos tipos de táboas en 3NF, que na práctica raramente se atopan, son afectadas por tales anomalías; estas son táboas que non satisfacen a forma normal de Boyce-Codd (FNBC ou BCNF\) ou, si satisfacen a FNBC, son insuficientes para satisfacer as formas normais máis altas 4FN ou 5FN.

O proceso de normalización ata a 3FN do exemplo mostra as seguintes dependencias funcionais entre atributos (DF):

* codProducto-> descricion.
* codProduto -> prezoUnidade.
* codCliente -> nomeCliente.
* codCliente -> cidade.
* CodPedido -> codCliente.
* codPedido -> cantidade.
* codPedido -> data.
* codPedido -> totalPedido.
* (codPedido, codProduto)-> cantidade.
* (codPedido, codProduto) -> total.

O esquema xeral xunto co seu contido en 3FN é:

Esquema3f.emf

Figura 2.22. Grafo relacional obtido o normalizar a 3FN

Taboas3FN.emf

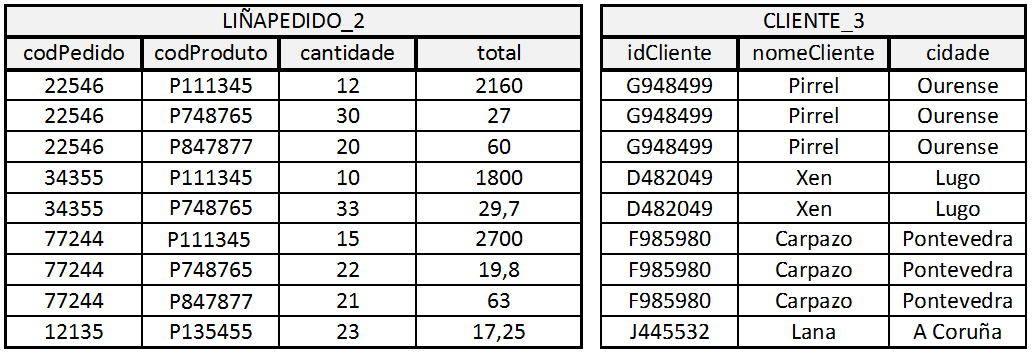


Figura.Táboas obtidas tras a normalización a 3FN

* 1. Forma normal de Boyce-Codd (FNBC ou BCNF). Dependencia de la clave

Un esquema R está en forma normal de Boyce\_Codd, si para toda dependencia X ->Y non trivial ( se Y non está contida en X) X é unha superclave de R, dito doutro xeito, non pode haber máis dependencias que con superclaves (o resto de atributos non clave da relación dependen funcionalmente dela). En termos menos formais, unha táboa está en FNBC si está en 3FN y os únicos determinantes son claves candidatas.

No noso exemplo se estudamos todas as relacións:

* PRODUTO\_2(codProduto, descricion,prezoUnidade).
* Clave:codProduto.
* DP: codProduto -> {descricion,prezoUnidade}.
* É superclaves atópase en FNBC.
* LIÑAPEDIDO\_2 (codPedido,CodProduto, cantidade, total).
* Clave: (codPedido, codProduto).
* DP: (codPedido, codProduto) -> {cantidade, total}.
* É superclaves atópase en FNBC.
* PEDIDO\_3 (codPedido,codCliente,data,totalPedido).
* Clave: (codPedido, codCliente).
* DP: (codPedido, codCliente-> {data, totalPedido}.
* É superclaves atópase en FNBC.
* CLIENTE\_3(codCliente,nomeCliente,cidade).
* Clave:codCliente.
* DP: codCliente -> {nomeCliente,cidade}.
* É superclaves atópase en FNBC.
* O exemplo xa se atopa en FNBC.

Estudemos agora un exemplo que non se atopa en FNBC pero si en 3FN.

TaboaEnderezoNONFNBC.emf

Figura . Táboa sen normalizar

O esquema relacional é ENDREZO (Rua, Numero, Piso, Municipio,Provincia,País, CP) onde:

* As claves poden ser: (Rua, Numero, Piso,CP) ou (Rua, Numero, Piso, Municipio, Provincia, Pais), non seleccionado esta última por non ser mínima.
* As dependencias funcionais serán:
* (Rua, Municipio, Provincia, Pais), -> CP; non é superclave xa que existen atributos independentes de ela como numero e piso.
* CP -> (Municipio, Provincia, Pais); non é superclave xa que existen atributos independentes de ela como numero, piso, e rúa.
* Polo tanto a relación non se atopa en FNBC pero si en en 3FN xa que aunque non son superclaves , si forman parte da clave.

FNBC é unha definición máis estrita da 3FN, afecta so ás relación que teñen claves solapadas, si R está en 3FN e só ten unha clave, ou ben R ten varias claves candidatas pero ningunha delas é composta, ou ben R ten varias claves compostas pero non solapadas entón R está en FNBC.