

NORMALIZACIÓN

El diseño de una BD relacional se puede realizar aplicando al mundo real, en una primera fase, un modelo como el modelo E/R, a fin de obtener un esquema conceptual; en una segunda fase, se transforma dicho esquema al modelo relacional mediante las correspondientes reglas de transformación.

También es posible, aunque quizás menos recomendable, obtener el esquema relacional sin realizar ese paso intermedio que es el esquema conceptual. En ambos casos, es conveniente (obligatorio en el modelo relacional directo) aplicar un conjunto de reglas, conocidas como Teoría de la normalización, que nos permiten asegurar que un esquema relacional cumple unas ciertas propiedades, evitando:

- La redundancia de los datos: repetición de datos en un sistema.
- Anomalías de actualización: inconsistencias de los datos como resultado de datos redundantes y actualizaciones parciales.
- Anomalías de borrado: pérdidas no intencionadas de datos debido a que se han borrado otros datos.
- Anomalías de inserción: imposibilidad de adicionar datos en la base de datos debido a la ausencia de otros datos.

En la práctica, si la BD se ha diseñado haciendo uso de modelos semánticos como el modelo E/R no suele ser necesaria la normalización. Por otro lado si nos proporcionan una base de datos creada sin realizar un diseño previo, es muy probable que necesitemos normalizar.

En la teoría de bases de datos relacionales, las formas normales (FN) proporcionan los criterios para determinar el grado de vulnerabilidad de una tabla a inconsistencias y anomalías lógicas.

Cuanto más alta sea la forma normal aplicable a una tabla, menos vulnerable será a inconsistencias y anomalías. Edgar F. Codd originalmente definió las tres primeras formas normales (1FN, 2FN, y 3FN) en 1970.

Estas formas normales se han resumido como requiriendo que todos los atributos sean atómicos, dependan de la clave completa y en forma directa (no transitiva). La forma normal de Boyce-Codd (FNBC) fue introducida en 1974 por los dos autores que aparecen en su denominación.

Las cuarta y quinta formas normales (4FN y 5FN) se ocupan específicamente de la representación de las relaciones muchos a muchos y uno a muchos entre los atributos y fueron introducidas por Fagin en 1977 y 1979 respectivamente.

Cada forma normal incluye a las anteriores.

La **normalización** es una técnica para diseñar la estructura lógica de los datos de un sistema de información en el modelo relacional, desarrollada por E. F. Codd en 1972. Es una estrategia de diseño de abajo arriba: se parte de los atributos y estos se van agrupando en relaciones (tablas) según su afinidad.

Aquí no se utilizara la normalización como una técnica de diseño de bases de datos, sino como una etapa posterior a la correspondencia entre el esquema conceptual y el esquema lógico, que elimine las dependencias no deseadas entre atributos. Las ventajas de la normalización son las siguientes:

- Evita anomalías en inserciones, modificaciones y borrados.
- Mejora la independencia de datos.
- No establece restricciones artificiales en la estructura de los datos.

Uno de los conceptos fundamentales en la normalización es el de *dependencia funcional*.

Una **dependencia funcional** es una relación entre atributos de una misma relación (tabla).

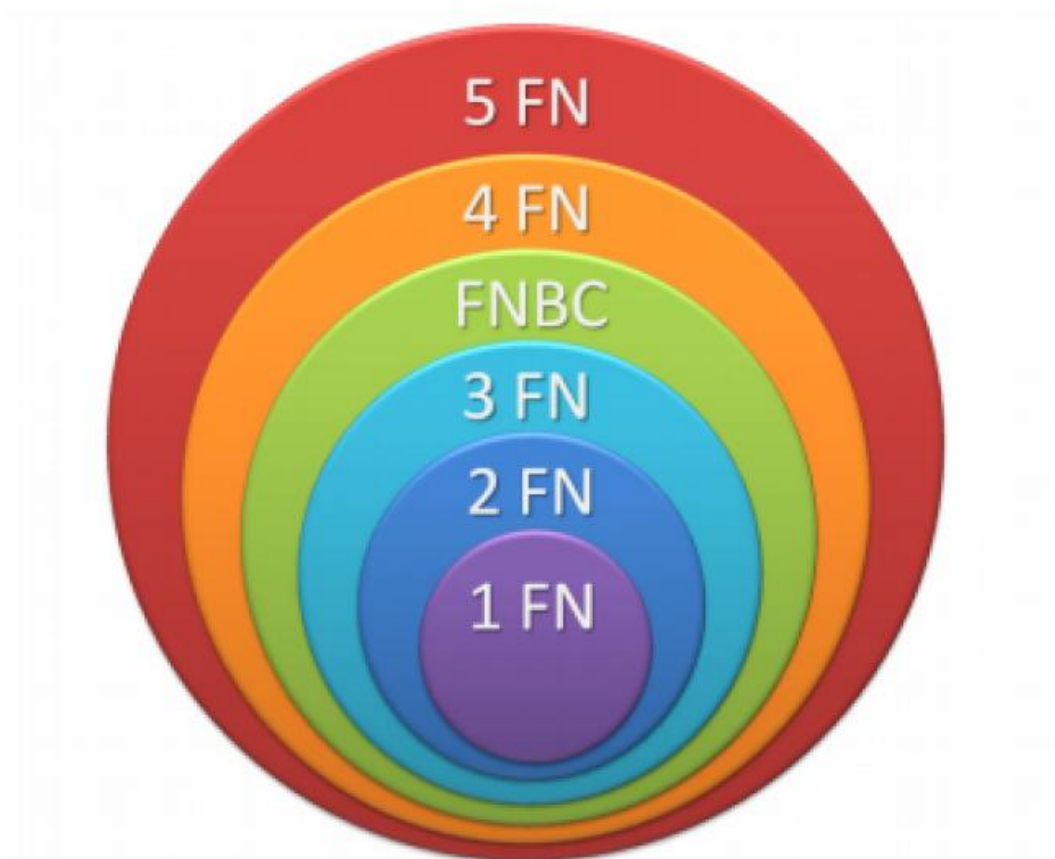
Las **dependencias** son propiedades inherentes al contenido semántico de los datos formando parte de las restricciones de usuario del modelo relacional. Entre los atributos de una relación pueden existir dependencias de varios tipos:

Dependencias funcionales

Son de primordial importancia a la hora de encontrar y eliminar la redundancia de los datos almacenados en las tablas de una base de datos relacional. Se centran en el estudio de las dependencias que presenta cada atributo de una relación con respecto al resto de atributos de la misma.

Dada una relación R que contiene los atributos X e Y se dice que Y depende funcionalmente

de X ($X \rightarrow Y$) si y solo si en todo momento cada valor de X tiene asociado un solo valor de Y. Esto es lo mismo que decir que si dos tuplas de R tienen el mismo valor para su atributo X forzosamente han de tener el mismo valor para el atributo Y.



Para un valor de A siempre aparece un valor de B. Ejemplo: Si A es el D.N.I., y B el Nombre, está claro que para un número de D.N.I., siempre aparece el mismo nombre de titular.

Dependencia funcional completa:

$A \rightarrow B$, si B depende de A en su totalidad.

Ejemplo: Tiene sentido plantearse este tipo de dependencia cuando A está compuesto por más de un atributo.

Por ejemplo, supongamos que A corresponde al atributo compuesto: D.N.I._Empleado + Cod._Dpto. y B es Nombre_Dpto. En este caso B depende del Cod_Dpto., pero no del D.N.I._Empleado. Por tanto no habría dependencia funcional completa.

Dependencia transitiva:

$A \rightarrow B \rightarrow C$. Si $A \rightarrow B$ y $B \rightarrow C$,

Entonces decimos que C depende de forma transitiva de A.

Ejemplo: Sea A el D.N.I. de un alumno, B la localidad en la que vive y C la provincia. Es un caso de dependencia transitiva $A \rightarrow B \rightarrow C$.

Determinante funcional:

Todo atributo, o conjunto de ellos, de los que depende algún otro atributo.

Ejemplo:

El D.N.I. es un determinante funcional pues atributos como nombre, dirección, localidad, etc, dependen de él.

Dependencia multivaluada:

$A \twoheadrightarrow B$.

Son un tipo de dependencias en las que un determinante funcional no implica un único valor, sino un conjunto de ellos.

Un valor de A siempre implica varios valores de B.

Ejemplo:

CursoBachillerato \twoheadrightarrow Modalidad. Para primer curso siempre va a aparecer en el campo Modalidad uno de los siguientes valores: Ciencias, Humanidades/Ciencias Sociales o Artes. Igual para segundo curso.

Primera Forma Normal: 1FN

Una Relación está en 1FN si y sólo si cada atributo es atómico.

Ejemplo: Supongamos que tenemos la siguiente tabla con datos de alumnado de un centro de enseñanza secundaria.

DNI	Nom-bre	Cur-so	FechaMa-trícula	Tutor	LocalidadA-lumno	ProvinciaA-lumno	Teléfonos
11111111	AEva	1ESO-A	01-Julio-2016	Isabel	Écija	Sevilla	660111222
22222222	BAAna	1ESO-A	09-Julio-2016	Isabel	Écija	Sevilla	660222333 660333444 660444555
33333333	CSusa-na	1ESO-B	11-Julio-2016	Ro-berto	Écija	Sevilla	
44444444	DJuan	2ESO-A	05-Julio-2016	Fede-rico	El Villar	Córdoba	
55555555	EJosé	2ESO-A	02-Julio-2016	Fede-rico	El Villar	Córdoba	661000111 661000222

Como se puede observar, esta tabla no está en 1FN puesto que el campo Teléfonos contiene varios datos dentro de una misma celda y por tanto no es un campo cuyos valores sean atómicos.

La solución sería la siguiente:

DNI	Nombre	Curso	FechaMatrícula	Tutor	LocalidadAlumno	ProvinciaAlumno
11111111A	Eva	1ESO-A	01-Julio-2016	Isabel	Écija	Sevilla
22222222B	Ana	1ESO-A	09-Julio-2016	Isabel	Écija	Sevilla
33333333C	Susana	1ESO-B	11-Julio-2016	Roberto	Écija	Sevilla
44444444D	Juan	2ESO-A	05-Julio-2016	Federico	El Villar	Córdoba
55555555E	José	2ESO-A	02-Julio-2016	Federico	El Villar	Córdoba

DNI	Teléfono
11111111A	660111222
22222222B	660222333
22222222B	660333444
22222222B	660444555
55555555E	661000111
55555555E	661000222

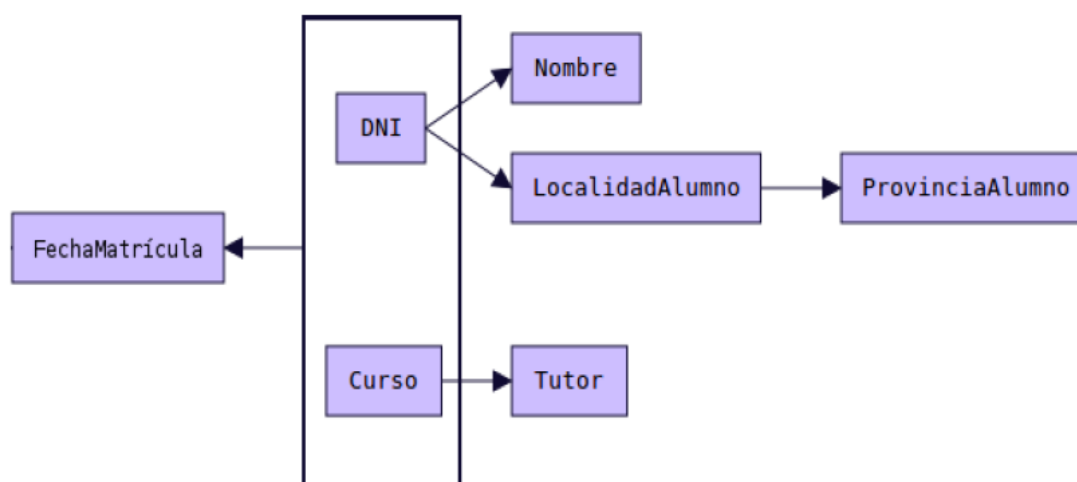
Segunda Forma Normal: 2FN

Una Relación esta en 2FN si y sólo si está en 1FN y todos los atributos que no forman parte de la Clave Principal tienen dependencia funcional completa de ella.

Seguimos con el ejemplo anterior. Trabajaremos con la siguiente tabla:

DNI	Nombre	Curso	FechaMatrícula	Tutor	LocalidadAlumno	ProvinciaAlumno
11111111A	Eva	1ESO-A	01-Julio-2016	Isabel	Écija	Sevilla
22222222B	Ana	1ESO-A	09-Julio-2016	Isabel	Écija	Sevilla
33333333C	Susana	1ESO-B	11-Julio-2016	Roberto	Écija	Sevilla
44444444D	Juan	2ESO-A	05-Julio-2016	Federico	El Villar	Córdoba
55555555E	José	2ESO-A	02-Julio-2016	Federico	El Villar	Córdoba

Vamos a examinar las dependencias funcionales. El gráfico que las representa es el siguiente:



Siempre que aparece un DNI aparecerá el Nombre correspondiente y la LocalidadAlumno correspondiente.

Por tanto $\text{DNI} \rightarrow \text{Nombre}$ y $\text{DNI} \rightarrow \text{LocalidadAlumno}$.

Por otro lado siempre que aparece un Curso aparecerá el Tutor correspondiente. Por tanto $\text{Curso} \rightarrow \text{Tutor}$.

Los atributos Nombre y LocalidadAlumno no dependen funcionalmente de Curso, y el atributo Tutor no depende funcionalmente de DNI.

El único atributo que sí depende de forma completa de la clave compuesta DNI y Curso es FechaMatrícula:

$(\text{DNI}, \text{Curso}) \rightarrow \text{FechaMatrícula}$.

A la hora de establecer la Clave Primaria de una tabla debemos escoger un atributo o conjunto de ellos de los que dependan funcionalmente el resto de atributos.

Además debe ser una dependencia funcional completa. Si escogemos DNI como clave primaria, tenemos un atributo (Tutor) que no depende funcionalmente de él. Si escogemos Curso como clave primaria, tenemos otros atributos que no dependen de él.

Si escogemos la combinación (DNI, Curso) como clave primaria, entonces sí tenemos todo el resto de atributos con dependencia funcional respecto a esta clave.

Pero es una dependencia parcial, no total (salvo FechaMatrícula, donde sí existe dependencia completa).

Por tanto esta tabla no está en 2FN. La solución sería la siguiente:

Alumnos:

DNI	Nombre	Localidad	Provincia
11111111A	Eva	Écija	Sevilla
22222222B	Ana	Écija	Sevilla
33333333C	Susana	El Villar	Córdoba
44444444D	Juan	El Villar	Córdoba
55555555E	José	Écija	Sevilla

Matrículas:

DNI	Curso	FechaMatrícula
11111111A	1ESO-A	01-Julio-2016
22222222B	1ESO-A	09-Julio-2016
33333333C	1ESO-B	11-Julio-2016
44444444D	2ESO-A	05-Julio-2016
55555555E	2ESO-A	02-Julio-2016

Cursos:

Curso	Tutor
1ESO-A	Isabel
1ESO-B	Roberto
2ESO-A	Federico

Tercera Forma Normal: 3FN

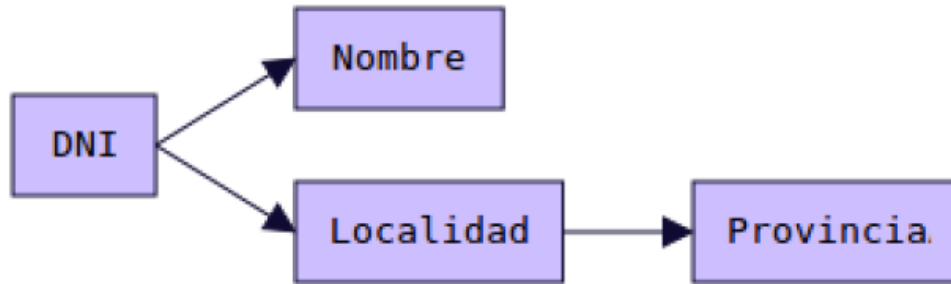
Una Relación esta en 3FN si y sólo si está en 2FN y no existen dependencias transitivas. Todas las dependencias funcionales deben ser respecto a la clave principal.

Seguimos con el ejemplo anterior. Trabajaremos con la siguiente tabla:

Alumnos:

DNI	Nombre	Localidad	Provincia
11111111A	Eva	Écija	Sevilla
22222222B	Ana	Écija	Sevilla
33333333C	Susana	El Villar	Córdoba
44444444D	Juan	El Villar	Córdoba
55555555E	José	Écija	Sevilla

Las dependencias funcionales existentes son las siguientes. Como podemos observar existe una dependencia funcional transitiva: DNI→ Localidad→Provincia



Para que la tabla esté en 3FN, no pueden existir dependencias funcionales transitivas. Para solucionar el problema deberemos crear una nueva tabla.

El resultado es:

Alumnos:

DNI	Nombre	Localidad
11111111A	Eva	Écija
22222222B	Ana	Écija
33333333C	Susana	El Villar
44444444D	Juan	El Villar
55555555E	José	Écija

Localidades:

Localidad	Provincia
Écija	Sevilla
El Villar	Córdoba

RESULTADO FINAL

Alumnos:

DNI	Nombre	Localidad
11111111A	Eva	Écija
22222222B	Ana	Écija
33333333C	Susana	El Villar
44444444D	Juan	El Villar
55555555E	José	Écija

Localidades

Localidad	Provincia
Écija	Sevilla
El Villar	Córdoba

Teléfonos

DNI	Teléfono
11111111A	660111222
22222222B	660222333
22222222B	660333444
22222222B	660444555
55555555E	661000111
55555555E	661000222

Matrículas

DNI	Curso	FechaMatrícula
11111111A	1ESO-A	01-Julio-2016
22222222B	1ESO-A	09-Julio-2016
33333333C	1ESO-B	11-Julio-2016
44444444D	2ESO-A	05-Julio-2016
55555555E	2ESO-A	02-Julio-2016

Cursos

Curso	Tutor
1ESO-A	Isabel
1ESO-B	Roberto
2ESO-A	Federico

Forma Normal de Boyce-Codd: FNBC

Una Relación esta en FNBC si está en 3FN y no existe solapamiento de claves candidatas.

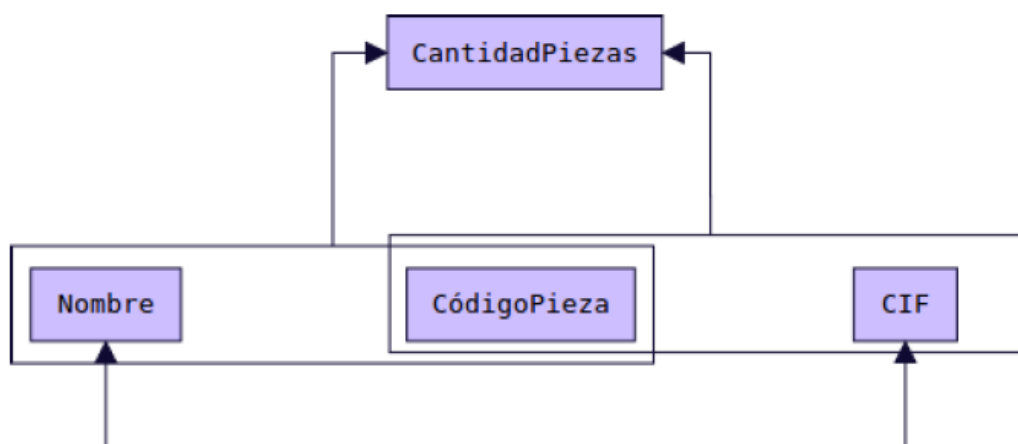
Solamente hemos de tener en cuenta esta forma normal cuando tenemos varias claves candidatas compuestas y existe solapamiento entre ellas.

Ejemplo: Tenemos una tabla con información de proveedores, códigos de piezas y cantidades de esa pieza que proporcionan los proveedores. Cada proveedor tiene un nombre único. Los datos son:

Suministros

CIF	Nombre	CódigoPieza	CantidadPiezas
S-11111111A	Ferroman	1	10
B-22222222B	Ferrotex	1	7
M-33333333C	Ferropet	3	4
S-11111111A	Ferroman	2	20
S-11111111A	Ferroman	3	15
B-22222222B	Ferrotex	2	8
B-22222222B	Ferrotex	3	4

El gráfico de dependencias funcionales es el siguiente:



El atributo CantidadPiezas tiene dependencia funcional de dos claves candidatas compuestas, que son:

(NombreProveedor, CódigoPieza)

(CIFProveedor, CódigoPieza)

Existe también una dependencia funcional en doble sentido (que no nos afecta): NombreProveedor <-> CIFProveedor.

Para esta tabla existe un solapamiento de 2 claves candidatas compuestas.

Para evitar el solapamiento de claves candidatas dividimos la tabla.

La solución es:

Proveedores:

CIF	Nombre
S-11111111A	Ferroman
B-22222222B	Ferrotex
M-33333333C	Ferropet

Suministros:

CIF	CódigoPieza	CantidadPiezas
S-11111111A	1	10
B-22222222B	1	7
M-33333333C	3	4
S-11111111A	2	20
S-11111111A	3	15
B-22222222B	2	8
B-22222222B	3	4

Otro ejemplo:

DNI	NÚM_SEG_SOC	NOMBRE	APELLIDOS	DEPARTAMENTO	PUESTO	SALARIO
413245-B	28-1234566	JUAN	RAMOS	COMPRAS	GERENTE	2.300
23456-J	28-2345686	PEDRO	PÉREZ	NÓMINAS	AUXILIAR	1.200
123123-C	19-458766	MARÍA	GIL	ALMACÉN	CONSERJE	1.530
123456-B	45-223344	ANTONIO	SANZ	COMPRAS	GESTIÓN	2.200

Esta relación se transforma en dos tablas: una contendrá la clave junto con las claves candidatas

(EMPLEADOS) y la otra la clave con el resto de campos (EMPLE_TRABAJO),

DNI	NUM_SEG_SOC	NOMBRE	APELLIDOS
413245-B	28-1234566	JUAN	RAMOS
23456-J	28-2345686	PEDRO	PÉREZ
123123-C	19-458766	MARÍA	GIL
1234556-B	45-223344	ANTONIO	SANZ

DNI (FK)	DEPARTAMENTO	PUESTO	SALARIO
413245-B	COMPRAS	GERENTE	2.300
23456-J	NÓMINAS	AUXILIAR	1.200
123123-C	ALMACÉN	CONSERJE	1.530
1234556-B	COMPRAS	GESTIÓN	2.200

Cuarta Forma Normal: 4FN

Una Relación esta en 4FN si y sólo si está en 3FN (o FNBC) y las únicas dependencias multivaluadas son aquellas que dependen de las claves candidatas.

Ejemplo: Tenemos una tabla con la información de nuestros alumnos y alumnas y las asignaturas que cursan así como los deportes que practican.

Alumnos:

Estudiante	Asignatura	Deporte
1111111A	Matemáticas, Lengua	Baloncesto
2222222B	Matemáticas	Fútbol, Natación

Estudiante	Asignatura	Deporte
1111111A	Matemáticas	Natación
1111111A	Matemáticas	Baloncesto
1111111A	Lengua	Natación
1111111A	Lengua	Baloncesto
2222222B	Matemáticas	Fútbol
2222222B	Matemáticas	Natación

Para normalizar esta tabla, debemos darnos cuenta que la oferta de asignaturas está compuesta por un conjunto de valores limitado. Igual sucede con los deportes. Por tanto existen dos dependencias multivaluadas:

Estudiante→→ Asignatura

Estudiante→→ Deporte

Por otro lado no existe ninguna dependencia entre la asignatura cursada y el deporte practicado. Para normalizar a 4FN creamos 2 tablas:

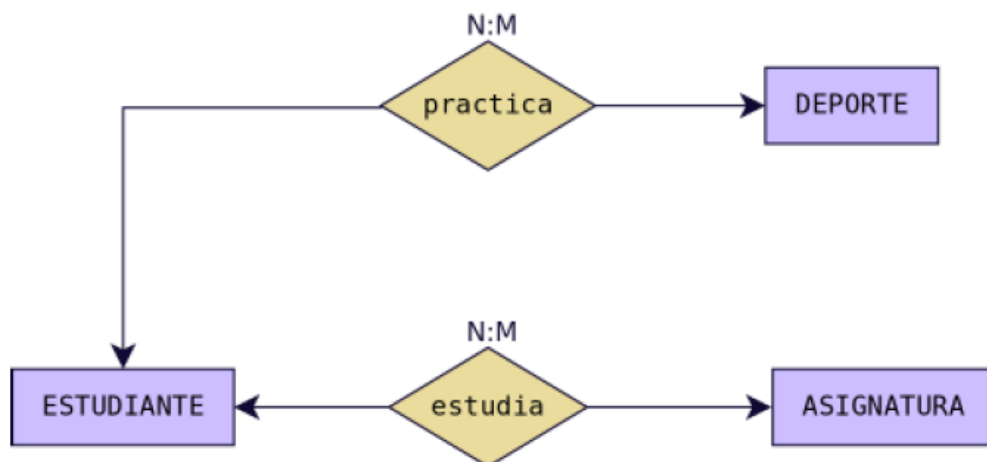
Estudia asignatura:

Estudiante	Asignatura
11111111A	Matemáticas
11111111A	Lengua
22222222B	Matemáticas

Practica deporte:

Estudiante	Deporte
11111111A	Natación
11111111A	Baloncesto
22222222B	Fútbol
22222222B	Natación

Esquema E/R



Quinta Forma Normal: 5FN

La quinta forma normal (5FN), es una generalización de la anterior. También conocida como forma normal de proyección-únion (PJ/NF).

Una tabla se dice que está en 5NF si y sólo si está en 4NF y cada dependencia de unión (join) en ella es implicada por las claves candidatas.

Ejemplo: Tenemos una tabla con varios proveedores que nos proporcionan piezas para distintos proyectos.

Asumimos que un Proveedor suministra ciertas Piezas en particular, un Proyecto usa ciertas Piezas, y un Proyecto es suplido por ciertos Proveedores, entonces tenemos las siguientes dependencias multivaluadas:

Proveedor →→ Pieza

Pieza →→ Proyecto

Proyecto →→ Proveedor

Se puede observar como se produce un ciclo:

Proveedor →→ Pieza →→ Proyecto →→ Proveedor
(nuevamente)

Proveedor:

Proveedor	Pieza	Proyecto
E1, E4, E6	PI3, PI6	PR2, PR4
E2, E5	PI1, PI2	PR1, PR3
E3, E7	PI4, PI5	PR5, PR6

Suministros:

Proveedor	Pieza	Proyecto
E1	PI3	PR2
E1	PI3	PR4
E1	PI6	PR2
E1	PI6	PR4
E4	PI3	PR2
E4	PI3	PR4
E4	PI6	PR2
E4	PI6	PR4
E6	PI3	PR2
E6	PI3	PR4
E6	PI6	PR2
E6	PI6	PR4
E2	PI1	PR1
E2	PI1	PR3
E2	PI2	PR1
E2	PI2	PR3
E5	PI1	PR1
E5	PI1	PR3
E5	PI2	PR1
E5	PI2	PR3
E3	PI4	PR5
E3	PI4	PR6
E3	PI5	PR5
E3	PI5	PR6
E7	PI4	PR5
E7	PI4	PR6
E7	PI5	PR5
E7	PI5	PR6

Descomponemos la tabla en 3 tabla nuevas: Proveedor-Pieza, Pieza-Proyecto, Proyecto-Proveedor.

Proveedor/pieza:

Proveedor	Pieza
E1	PI3
E1	PI6
E4	PI3
E4	PI6
E6	PI3
E6	PI6
E2	PI1
E2	PI2
E5	PI1
E5	PI2
E3	PI4
E3	PI5
E7	PI4
E7	PI5

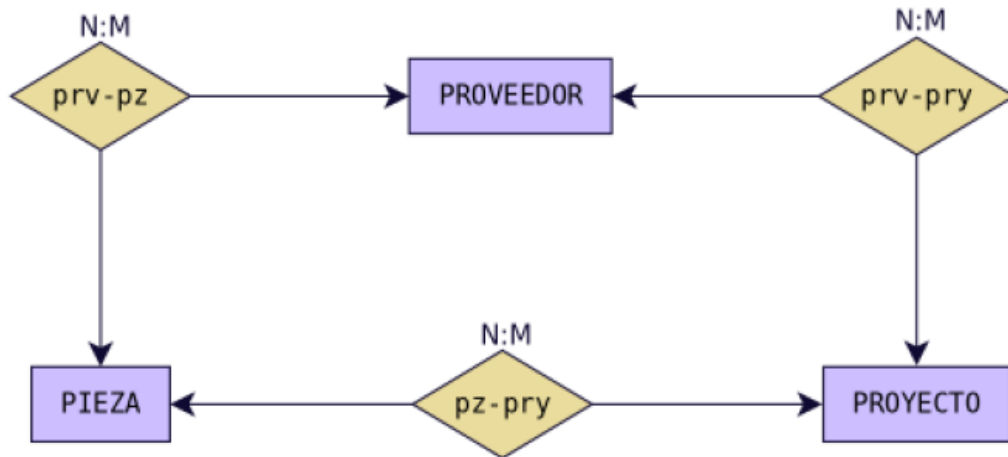
Pieza/proyecto:

Pieza	Proyecto
PI3	PR2
PI3	PR4
PI6	PR2
PI6	PR4
PI1	PR1
PI1	PR3
PI2	PR1
PI2	PR3
PI4	PR5
PI4	PR6
PI5	PR5
PI5	PR6

Proyecto/proveedor:

Proyecto	Proveedor
PR2	E1
PR4	E1
PR2	E4
PR4	E4
PR2	E6
PR4	E6
PR1	E2
PR3	E2
PR1	E5
PR3	E5
PR5	E3
PR6	E3
PR5	E7
PR6	E7

El producto natural de estas 3 tablas nos da la tabla original.
Proveedor-Pieza |x| Pieza-Proyecto |x| Proyecto-Proveedor
= Suministros
Diagrama E/R equivalente



Conceptos

A continuación vamos a definir una serie de conceptos fundamentales para comprender las formas normales que explicaremos después:

Dependencia Funcional: Es una conexión entre uno o más atributos

DNI \rightarrow Nombre y apellidos

Dependencia Funcional Reflexiva: Si "Y" está incluido en "X" entonces $X \rightarrow Y$

Si dirección y nombre están incluidos en DNI entonces con el DNI se puede recuperar la dirección y el nombre

Dependencia Funcional Aumentativa: Si $X \rightarrow Y$ entonces $XZ \rightarrow YZ$

DNI \rightarrow Nombre

DNI, Dirección \rightarrow Nombre, Dirección

Dependencia Funcional Transitiva: Si $X \rightarrow Y \rightarrow Z$ entonces $X \rightarrow Z$

Fecha de nacimiento \rightarrow Edad

Edad \rightarrow Conducir

Fecha de nacimiento \rightarrow Edad \rightarrow Conducir

Ejemplo:

Para explicar las formas normales vamos a poner una tabla de ejemplo de Empleados. La siguiente tabla muestra la información de una empresa cuyos puestos de trabajo están regulados por el Estado, de modo que el salario de cada empleado depende del puesto. Datos empleados: ID, nombre, puesto salario e email, siendo ID la clave primaria.

ID (Pk)	Nombre	Puesto	Salario	Emails
1	Juan Pérez	Jefe de área	3000	juan@test.com;jefe1@test.com
2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com;admin@test.com

Tabla 1. Empleados

1FN

Una tabla está en primera forma normal si:

- Sus atributos contienen valores atómicos (esto quiere decir que tienen que ser indivisibles)

En el ejemplo podemos ver que no se cumple para el atributo «Emails»

ID (Pk)	Nombre	Puesto	Salario	Emails	
1	Juan Pérez	Jefe de área	3000	juan@test.com;jefe1@test.com	X
2	José Sánchez	Administrativo	1500	jsanchez@test.com	
3	Ana Díaz	Administrativo	1500	adiaz@test.com;admin@test.com	X

Tabla 1. Empleados

Para solucionarlo existen 2 opciones:

1. Duplicar registros con valores repetidos:

- Se elimina el atributo «Emails» que violaba la condición
- Se incluye un nuevo atributo «Email» que sí sea indivisible. Por lo que se crea una nueva clave primaria con este nuevo atributo

La nueva clave primaria será «ID-Email»

ID (Pk)	Nombre	Puesto	Salario	Emails (Pk)
1	Juan Pérez	Jefe de área	3000	juan@test.com
1	Juan Pérez	Jefe de área	3000	jefe1@test.com

2	José Sánchez	Administrativo	1500	jsanchez@test.com
3	Ana Díaz	Administrativo	1500	adiaz@test.com
3	Ana Díaz	Administrativo	1500	admin@test.com

Tabla 2. Empleados (a)

2. Separar atributo «Email» en otra tabla:

- Se crea una nueva tabla Empleados (b) que no contenga el atributo «Email»

ID (Pk)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500

Tabla 3. Empleados (b)

- Se crea una nueva tabla EMAILS con clave primaria ID-Email. Las tablas Emails y Empleados se relacionan por el campo ID

ID	Emails (Pk)
1	juan@test.com
1	jefe1@test.com
2	jsanchez@test.com
3	adiaz@test.com
3	admin@test.com

Tabla 4. Emails

2FN

Una tabla está en 2FN si:

- Está en 1FN
- Todos los atributos que no son clave primaria tienen dependencia funcional completa con respecto a todas las claves existentes en el esquema. Para recuperar un

atributo no clave, se necesita acceder por la clave completa, no por una subclave

- Las 2FN aplican a las relaciones con claves primarias compuestas por dos o más atributos

ID (Pk)	Emails (Pk)	Nombre	Puesto	Salario
1	juan@test.com	Juan Pérez	Jefe de área	3000
1	jefe1@test.com	Juan Pérez	Jefe de área	3000
2	jsanchez@test.com	José Sánchez	Administrativo	1500
3	adiaz@test.com	Ana Díaz	Administrativo	1500
3	admin@test.com	Ana Díaz	Administrativo	1500

Tabla 2. Empleados (a)

En la tabla de Empleados (a) se pueden ver las dependencias de los atributos:

ID → Nombre, puesto, salario

Puesto → Salario

Observamos que los atributos nombre, puesto y salario dependen únicamente del campo ID, por lo que no cumple la 2FN.

Para solucionarlo:

→ Actuar sobre los atributos con dependencias incompletas:

- Eliminar los atributos con dependencias incompletas
- Crear nueva tabla con los atributos y la clave de la que depende

ID (Pk)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500

Tabla 3. Empleados (b)

ID	Emails (Pk)
----	-------------

1	juan@test.com
1	jefe1@test.com
2	jsanchez@test.com
3	adiaz@test.com
3	admin@test.com

Tabla 4. Emails

Se llega a la misma solución que con la 1FN.

3FN

Una tabla está en 3FN si:

- Está en 2FN
- Todos los atributos que no son clave primaria no dependen transitivamente de ésta

Por tanto hay que buscar dependencias funcionales entre atributos que no estén en la clave.

ID (Pk)	Nombre	Puesto	Salario
1	Juan Pérez	Jefe de área	3000
2	José Sánchez	Administrativo	1500
3	Ana Díaz	Administrativo	1500

Tabla 3. Empleados (b)

Las dependencias transitivas son:

ID → Puesto

Puesto → Salario

Observamos como la dependencia Puesto – Salario tiene dependencia transitiva con la clave primaria.

Para solucionarlo:

→ Actuar sobre los atributos con dependencias transitivas

- Separar en una tabla adicional los atributos que tienen dependencia transitiva con la clave (Salario) y establecer como Pk el campo que define la transitividad (Puesto)

Puesto (Pk)	Salario
Jefe de área	3000
Administrativo	1500

Tabla 5. Puestos

ID (Pk)	Nombre	Puesto (fk)
1	Juan Pérez	Jefe de área
2	José Sánchez	Administrativo
3	Ana Díaz	Administrativo

Tabla 6. Empleados (c)

- Se añade el campo «Puesto» como *Foreign Key*:

Primera forma normal(1FN)

No se debe tener lista de atributos en un único campo

CÓDIGO	NOMBRE	APELLIDO	PROFESIONES
0001	VICTOR	DUQUE	Asistente-jardinero-pintor
002	JOSÉ	MELO	Pintor-albañil-todero
003	CARLOS	TUN	Docente-Conductor

Primera forma normal(1FN)

No se debe tener lista de atributos en un único campo

CÓDIGO	NOMBRE	APELLIDO	PROFESIONE1	PROFESIONE2	PROFESIONE3
0001	VICTOR	DUQUE	Asistente-	jardinero	Pintor
002	JOSÉ	MELO	Pintor-	albañil	todero
003	CARLOS	TUN	Docente-	Conductor	

Primera forma normal(1FN)

CÓDIGO	NOMBRE	APELLIDO	CODIGO PROFESION	PROFESIONES
0001	VICTOR	DUQUE	P01	Asistente
0001	VICTOR	DUQUE	P02	Jardinero
0001	VICTOR	DUQUE	P03	Pintor
002	JOSÉ	MELO	P03	Pintor
002	JOSÉ	MELO	P04	Albañil
002	JOSÉ	MELO	P05	Todero
003	CARLOS	TUN	P06	Docente
003	CARLOS	TUN	P07	Conductor

BASES DE DATOS

Primera forma normal(1FN)

CÓDIGO	NOMBRE	APELLIDO
001	VICTOR	DUQUE
002	JOSÉ	MELO
003	CARLOS	TUN



PERSONA_PROFESION

CÓDIGO	CodProfesion	Profesión
001	P01	Asistente
001	P02	Jardinero
001	P03	Pintor
002	P03	Pintor
002	P04	Albañil
002	P05	Todero
003	P06	Docente
003	P07	Conductor
003	P08	Investigador

EVITAR ERRORES DE ACTUALIZACIÓN

Segunda forma normal(2FN)

TODOS LOS ATRIBUTOS DEBEN
DEPENDER DIRECTAMENTE DE LA
LLAVE PRIMARIA

BASES DE DATOS

Segunda forma Normal(2FN)

CÓDIGO	NOMBRE	APELLIDO
001	VICTOR	DUQUE
002	JOSÉ	MELO
003	CARLOS	TUN



PERSONA_PROFESION		
CÓDIGO	CodProfesion	Profesión
001	P01	Asistente
001	P02	Jardinero
001	P03	Pintor
002	P03	Pintor
002	P04	Albañil
002	P05	Todero
003	P06	Docente
003	P07	Conductor
003	P08	Investigador

EVITAR ERRORES DE ACTUALIZACIÓN

BASES DE DATOS

Segunda forma Normal(2FN)

PROFESION

CÓDIGO	NOMBRE
P01	ASISTENTE
P02	JARDINERO
P03	PINTOR
P04	ALBAÑIL
P05	TODERO
P06	DOCENTE
P07	CONDUCTOR
P08	INVESTIGADOR

PERSONA_PROFESION

CÓDIGO	codprofesion
001	P01
001	P02
001	P03
002	P03
002	P04
002	P05
003	P06
003	P07

PERSONA

CÓDIGO	NOMBRE	APELLIDO
001	VICTOR	DUQUE
002	JOSÉ	MELO
003	CARLOS	TUN

Tercera forma normal(3FN)

No Dependencias Transitivas

Tercera forma normal(3FN)

Evitar las relaciones transitivas

$A \rightarrow B$, $A \rightarrow C$, $C \rightarrow Z$

Código	Nombre	Apellido	CodCargo	NombreCargo
C001	Carlos	Kumico	Car01	Auxiliar
C002	Juan	Perez	Car01	Auxiliar
C003	Betty	Ordoñez	Car02	Jefe Depto



Tercera forma normal(3FN)

Evitar las relaciones transitivas

$A \rightarrow B, A \rightarrow C, C \rightarrow Z$

Código	Nombre	Apellido	CodCargo
C001	Carlos	Kumico	Car01
C002	Juan	Perez	Car01
C003	Betty	Ordoñez	Car02

Cargos

CodCargo	Cargo
Car01	Auxiliar
Car02	JefeDepto

