



Análisis y Diseño Lógico de Sistemas

3419B201-A





Normalización

- La Normalización, abarca dos tópicos:
 - **Dependencia Funcional:** Técnica de diseño que permite examinar las relaciones entre los atributos.
 - **Formas Normales:** Pruebas para el agrupamiento óptimo de los atributos.



Con la
normalización
se pretende
que:

- Los atributos con una relación lógica fuerte (dependencia funcional) se encuentren en la misma relación.
- Se defina el número mínimo de atributos necesarios para soportar requisitos de datos de una organización.
- La información tengan una redundancia mínima. Cada atributo se representa una sola vez, con excepción de las claves foráneas.
 - Actualización con un mínimo de operaciones.
 - Reduce posibles inconsistencias de datos.
 - Reduce espacio de almacenamiento.



Justificación

- Este es un ejemplo muy sencillo, un esquema de empleados que trabajan en proyectos, en una relación muchos a muchos.
- `EMPLEADO(RUT, nombre)`
 - Clave primaria: **RUT**
- `PROYECTO(código, presupuesto, inicio, final, ciudad)`
 - Clave primaria: **código**
- `TRABAJAR(RUT, código, desde, hasta)`
 - Clave primaria: (RUT, código, desde)
 - Clave foránea: **RUT** → EMPLEADO
 - Clave foránea: **código** → PROYECTO



Justificación



Empleado

<u>rut</u>	<u>nombre</u>
22.123.987-K	Miguel Strogoff
12.321.687-K	Hermann Hesse

Proyecto

<u>código</u>	<u>presupuesto</u>	<u>inicio</u>	<u>final</u>	<u>ciudad</u>
P001-100	\$12.000000	06/06/2023	15/03/2024	Valdivia

Trabajar

<u>rut</u>	<u>código</u>	<u>desde</u>	<u>hasta</u>
22.123.987-K	P001-100	06/06/2023	15/03/2024
12.321.687-K	P001-100	02/01/2024	15/03/2024

Cliente

<u>rut</u>	<u>nombre</u>
22.123.987-K	Miguel Strogoff
12.321.687-K	Hermann Hesse
12.321.687-K	Hermann Hesse

Justificación

- Queremos añadir la información de “TELÉFONO DE EMPLEADO”: ¿dónde pongo esa nueva columna? ¿En EMPLEADO, en PROYECTO o en TRABAJAR? La respuesta es obvia pero vamos a suponer, a continuación, que lo hacemos mal.
- Estamos asumiendo, también, que el teléfono para un empleado X es único.



Justificación



- **EMPLEADO**(RUT, nombre)
- Clave primaria: RUT
- **PROYECTO**(código, presupuesto, inicio, final, ciudad)
- Clave primaria: código

- **TRABAJAR**(RUT, teléfono, código, desde, hasta)
- Clave primaria: (RUT, código, desde)
- Clave foránea: RUT → EMPLEADO
- Clave foránea: código → PROYECTO

Justificación

- Anomalías de actualización
 - Efectos no deseados en el manejo de la información almacenada

RUT	TELÉFONO	CÓDIGO	DESDE	HASTA
22446688-K	5632224	A111	Enero	febrero
22446688-K	5632224	A112	Enero	febrero
22446688-K	5632224	A112	Febrero	abril
22446688-K	5632224	A116	Enero	marzo
11116666-0	52111111	A112	Abril	Abril
11116666-0	52111111	A114	marzo	Abril

- Este sería el aspecto de la tabla TRABAJAR con algunos datos.
- Nunca olvidemos que la CP es (RUT, CÓDIGO, DESDE). A partir de aquí vamos a ver cuáles son los problemas con los que nos vamos a encontrar al operar con esta tabla: las ANOMALÍAS DE ACTUALIZACIÓN



Justificación

- Inserción = posibles inconsistencias
 - ¿Qué teléfono es el “bueno”?

<u>RUT</u>	TELÉFONO	<u>CÓDIGO</u>	<u>DESDE</u>	HASTA
22446688-K	5632224	A111	Enero	febrero
22446688-K	5632224	A112	Enero	febrero
22446688-K	5632224	A112	Febrero	abril
22446688-K	5632224	A116	Enero	marzo
11116666-0	52111111	A112	Abril	Abril
11116666-0	52111111	A114	marzo	Abril
22446688-K	6632224	A116	Marzo	abril

- Como he introducido **redundancia** puedo provocar una **inconsistencia de datos** y ahora no sé cuál es el teléfono correcto, si el que está en rojo o los anteriores.



Justificación

- Inserción = restricciones “artificiales”
 - ¿No puedo almacenar el teléfono de un empleado si no tiene proyecto?

<u>RUT</u>	TELÉFONO	<u>CÓDIGO</u>	<u>DESDE</u>	HASTA
22446688-K	5632224	A111	Enero	febrero
22446688-K	5632224	A112	Enero	febrero
22446688-K	5632224	A112	Febrero	abril
22446688-K	5632224	A116	Enero	marzo
11116666-0	5211111	A112	Abril	Abril
11116666-0	5211111	A114	marzo	Abril
22446688-K	6632224	A116	Marzo	Abril
33335555-7	5660626	(null)	(null)	(null)



Justificación

- Borrado = posible eliminación no deseada de información de la BD
 - Si elimino la relación del empleado con sus proyectos “pierdo” su teléfono.

RUT	TELÉFONO	CÓDIGO	DESDE	HASTA
22446688-K	5632224	A111	Enero	febrero
22446688-K	5632224	A112	Enero	febrero
22446688-K	5632224	A112	Febrero	abril
22446688-K	5632224	A116	Enero	marzo
11116666-0	5211111	A112	Abril	Abril
11116666-0	5211111	A114	marzo	Abril
22446688-K	6632224	A116	Marzo	Abril

- El empleado ya no está asignado a esos proyectos, se van a borrar esas 2 filas, pero es que, además, también elimino su teléfono: ¿yo quería perder el número de teléfono de ese empleado?



Justificación

- Modificación = costo computacional elevado
 - ...busca y reemplaza, busca y reemplaza...

RUT	TELÉFONO	CÓDIGO	DESDE	HASTA
22446688-K	5632224	A111	Enero	febrero
22446688-K	5632224	A112	Enero	febrero
22446688-K	5632224	A112	Febrero	abril
22446688-K	5632224	A116	Enero	marzo
11116666-0	5211111	A112	Abril	Abril
11116666-0	5211111	A114	marzo	Abril
22446688-K	6632224	A116	Marzo	Abril

- En teoría, si el empleado cambia de teléfono hay que buscar al empleado por toda la tabla y modificar todas las veces que aparece su teléfono y poner el nuevo.



Justificación



Ciente

<u>rut</u>	<u>nombre</u>	<u>fono</u>	<u>dirección</u>
22.123.987-K	Miguel Strogoff	+5694437689	Av. Los Leones Nro. 234, Depto 303

Ciente

<u>rut</u>	<u>nombre</u>	<u>fono</u>	<u>dirección</u>
22.123.987-K	Miguel Strogoff	+5694437689	Av. Los Leones Nro. 234, Depto 203
12.321.687-K	Hermann Hesse	+5694437689	Av. Holanda Nro. 591, Depto 901

Ciente

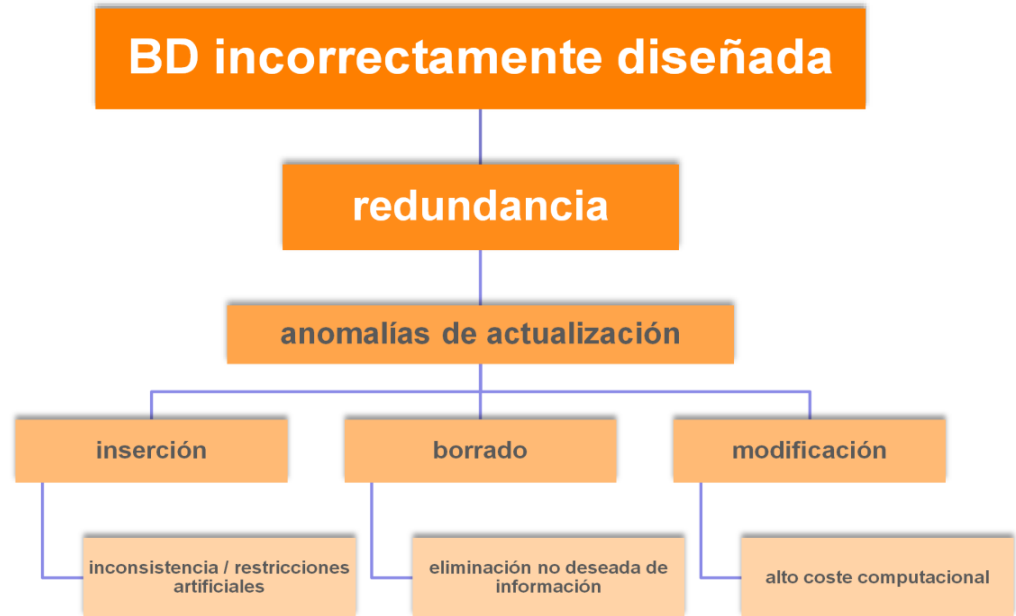
<u>rut</u>	<u>nombre</u>	<u>fono</u>	<u>dirección</u>
22.123.987-K	Miguel Strogoff	+5694437689	Av. Los Leones Nro. 234, Depto 203
12.321.687-K	Hermann Hesse	+5694437689, +5698765392	Av. Holanda Nro. 591, Depto 901

Ciente

<u>rut</u>	<u>nombre</u>	<u>fono</u>	<u>dirección</u>
22.123.987-K	Miguel Strogoff	+5694437689	Av. Los Leones Nro. 234, Depto 203
12.321.687-K	Hermann Hesse	+5694437689	Av. Holanda Nro. 591, Depto 901
12.321.687-K	Hermann Hesse	+5698765392	Av. Holanda Nro. 591, Depto 901

Justificación

- Resumiendo todo lo anterior, esto es el porqué de la necesidad de la **normalización**.



Justificación

- Solución: Normalización
 - Ya no como herramienta de diseño, sino como criterio de calidad.
 - Proceso de transformación de las tablas de una BDR hasta eliminar las anomalías de actualización
 - Estructura eficiente
 - Coherente con las especificaciones del sistema de información



Justificación

- **EMPLEADO**(RUT, nombre)
 - Clave primaria: RUT
- **PROYECTO**(código, presupuesto, inicio, final, ciudad)
 - Clave primaria: código
- **TRABAJAR**(RUT, ~~teléfono~~, código, desde, hasta)
 - Clave primaria: (RUT, código, desde)
 - Clave foránea: RUT → EMPLEADO
 - Clave foránea: código → PROYECTO
- El proceso de normalización, que veremos a continuación, nos diría que la columna TRABAJA.teléfono no está correctamente colocada, hay dependencias entre las columnas de la tabla que no estaban previstas en los requisitos originales del sistema de información.



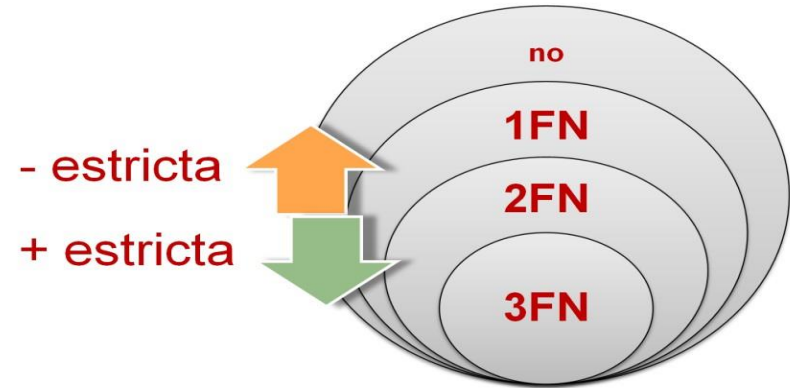
Justificación



- Normalizar
 - Correcta elección del esquema
 - Comprobación del esquema
- EMPLEADO(RUT, nombre, teléfono)
 - Clave primaria: RUT
- PROYECTO(código, presupuesto, inicio, final, ciudad)
 - Clave primaria: código
- TRABAJAR(RUT, código, desde, hasta)
 - Clave primaria: (RUT, código, desde)
 - Clave foránea: RUT → EMPLEADO
 - Clave foránea: código → PROYECTO

Justificación

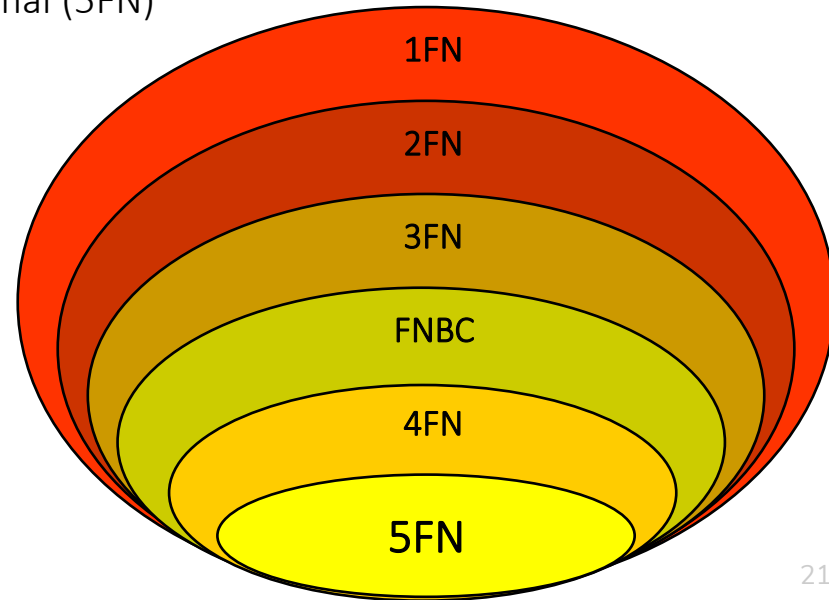
- Formas normales
 - Requisitos a cumplir por las tablas
 - Objetivo: llegar hasta 3FN
 - Si las tablas tienen una única Clave Candidata
 - Si tienen más FNBC? 4FN? 5FN?...



Formas Normales



- Primera, segunda y tercera Formas Normales
 - Codd 1970
- Forma Normal de Boyce y Codd (FNBC)
 - Boyce - Codd 1974
- Cuarta Forma Normal (4FN)
 - Fagin 1977
- Quinta Forma Normal (5FN)
 - Fagin 1979



Dependencia Funcional



- **DEF:** Sean α y β atributos de la relación R.
 - Decimos que α **determina funcionalmente a β** en R, denotado por $\alpha \rightarrow \beta$
 - También se puede decir que: β **depende funcionalmente de α**
 - Si y sólo si :
 - Para todos los pares de tuplas t1, t2 de la relación R, tales que $t1[\alpha] = t2[\alpha]$ también se cumple que $t1[\beta] = t2[\beta]$
- **Ejemplo:** cédula \rightarrow nombre.
- Cada vez que se tiene un número de cédula, esta debe coincidir con el mismo nombre.

Ejemplos de Dependencia Funcional



- Ejemplo de dependencia funcional en caso de datos de personas:
 - Rut \rightarrow nombre_persona, fecha_nacimiento
 - En cada ocasión que se encuentre un Rut, se encontrará asociado el mismo nombre y fecha de nacimiento.
 - Se lee: Nombre_persona y fecha_nacimiento **dependen funcionalmente** de rut.
 - También: rut **determina funcionalmente** a Nombre_persona y a fecha_nacimiento.

Ejemplos de Dependencia Funcional



- Sean las relaciones:
 - Préstamo (númeroPréstamo, sucursal, cliente, valor)
 - Cliente (cédula, nombreCliente, dirección, ciudad)
- Si NúmeroPréstamo \rightarrow cliente.
- Un préstamo sólo puede efectuarse a un cliente.
- Un cliente puede tener varios préstamos.
- NúmeroPréstamo \rightarrow valor, es cierta en préstamo?
- dirección \rightarrow ciudad, es cierta en cliente?

Ejemplos de Dependencia Funcional

- Ejemplo de dependencia funcional en caso de datos de un vehículo:
 - Patente → Marca, Modelo, Año_Fabricación
 - En caso de datos de salas de clases:
 - CódigoSala, Fecha, Hora → Curso, Profesor



Clave Candidata

- Sea K un conjunto de uno o más atributos de la relación R .
- **DEF:**
 - K es una clave candidata para la relación R si:
 - Si $K \rightarrow$ todos los atributos de R ;
 - Ningún subconjunto de K determina funcionalmente a todos los demás atributos de R .



Dependencia Funcional Completa



- **DEF:** Sean α y β atributos de la relación R. Decimos que β **depende funcionalmente de manera completa de α** Si y sólo si:
 - β depende funcionalmente de α pero no de ningún subconjunto propio de α .

Es decir, Una dependencia funcional $\alpha \rightarrow \beta$ es completa si la eliminación de cualquier atributo de α hace que la dependencia deje de existir.

- Cédula, nombre \rightarrow salario
- Si se quita el nombre la dependencia continúa Cédula \rightarrow salario
- Entonces no era completa

Axiomas de Armstrong



1. Reglas de reflexividad: (dependencia trivial)

Si α y β son conjuntos de atributos y $\beta \subseteq \alpha$, entonces se cumple que $\alpha \twoheadrightarrow \beta$.

Si los atributos (TipoDoc, número) de una persona son su ID, entonces con ellos podemos determinar el TipoDoc y el número.

2. Regla de aumento:

Si para los conjuntos de atributos α y β se cumple que $\alpha \twoheadrightarrow \beta$ y γ es un conjunto de atributos, entonces se cumple que $\alpha \gamma \twoheadrightarrow \beta \gamma$.
(cédula, teléfono) \twoheadrightarrow (nombre, teléfono)

3. Regla de la transitividad:

Si se cumple $\alpha \twoheadrightarrow \beta$ y se cumple $\beta \twoheadrightarrow \gamma$, entonces se cumple $\alpha \twoheadrightarrow \gamma$.

Reglas adicionales de Armstrong



Reglas adicionales, derivadas de las anteriores :

4. Reglas de unión:

Si se cumple $\alpha \rightarrow \beta$ y $\alpha \rightarrow \gamma$ se cumple $\alpha \rightarrow \beta \gamma$.

Cédula \rightarrow nombre y cédula \rightarrow teléfono

Cédula \rightarrow (nombre , teléfono)

5. Regla de la descomposición:

Si se cumple $\alpha \rightarrow \beta \gamma$ entonces se cumple $\alpha \rightarrow \beta$ y $\alpha \rightarrow \gamma$.

cédula \rightarrow (apellido, dirección)

cédula \rightarrow apellido y cédula \rightarrow dirección

6. Regla de la pseudo-transitividad:

Si $\alpha \rightarrow \beta$ y $\gamma \beta \rightarrow \delta$ entonces se cumple $\alpha \gamma \rightarrow \delta$.

Cédula \rightarrow Ciudad_residencia

(Teléfono, Ciudad_residencia) \rightarrow dirección_residencia

(Cédula,Teléfono) \rightarrow dirección_residencia

Implicación lógica de las D.F.



- **Ejemplo :**

Sea la relación $R(A, B, C, G, H, I)$

Con el conjunto de Dependencias Funcionales

$$F = \{ A \rightarrow B, A \rightarrow C, CG \rightarrow I, CG \rightarrow H, B \rightarrow H \}$$

Podemos encontrar nuevas dependencias funcionales implicadas lógicamente por F :

- | | | | |
|----|--|-----------------------------|---------------|
| a. | $A \rightarrow B$ y $B \rightarrow H$ | luego : $A \rightarrow H$. | por axioma-3. |
| b. | $CG \rightarrow H$ y $CG \rightarrow I$ | luego $CG \rightarrow HI$ | por axioma-4 |
| c. | $A \rightarrow C$ | luego $AG \rightarrow CG$ | por axioma-2 |
| d. | $AG \rightarrow CG$ y $CG \rightarrow I$ | luego $AG \rightarrow I$ | por axioma-3 |
| e. | $AG \rightarrow CG$ y $CG \rightarrow H$ | luego $AG \rightarrow H$ | por axioma-3. |

Dependencia Funcional



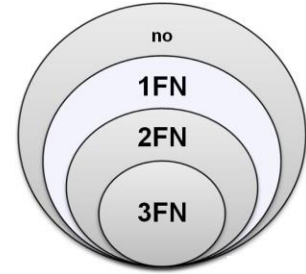
- “todo empleado puede tener un (único) teléfono”
 - RUT → teléfono
 - Por cada RUT sólo hay 1 valor de teléfono
 - (lo contrario, no necesariamente)

RUT	teléfono
21	965
24	
55	985
33	985

- En realidad, la definición anterior nos está diciendo que para cada valor de RUT, utilizando este ejemplo, solo hay un valor de teléfono: "*cada empleado no puede tener 2 o más teléfonos simultáneamente*".

Formas Normales

- Formas Normales
 - 1FN = estructura básica
 - 2FN = 1FN + DF completas
 - 3FN = 2FN + DF transitivas



- Vamos a definir las formas normales básicas. Hemos hablado de "reglas" a cumplir con cada forma normal. Estas primeras reglas tienen que ver con si las dependencias funcionales son completas o no, y con la existencia o no de dependencias funcionales transitivas. Así, hemos de definir qué es:
 - una dependencia funcional completa
 - una dependencia funcional transitiva

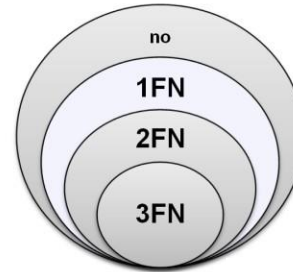


Formas Normales

- **1FN**

Una relación está en primera forma normal (1FN) si y sólo si todos los dominios simples subyacentes contienen sólo valores atómicos.

- No olvidemos las Claves Primarias!
 - Una Relación está en primera...

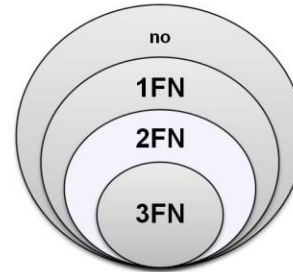


Formas Normales

- **2FN**

Una relación está en segunda forma normal (2FN) si y sólo si

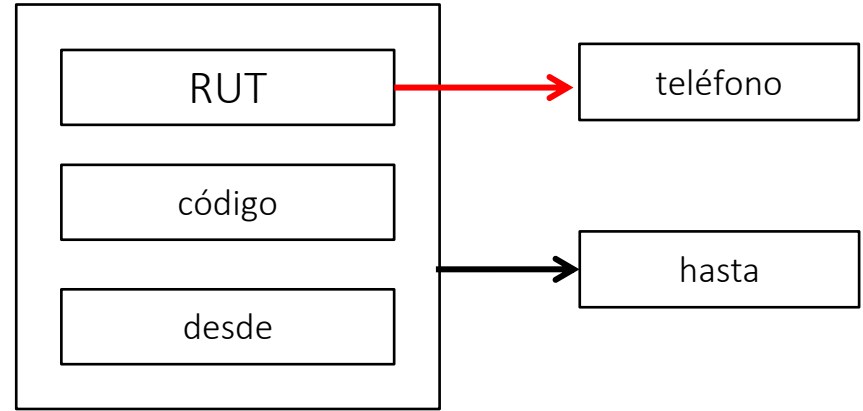
- Está en 1FN y
- todos los atributos no clave dependen de forma completa de **toda** clave candidata



Formas Normales



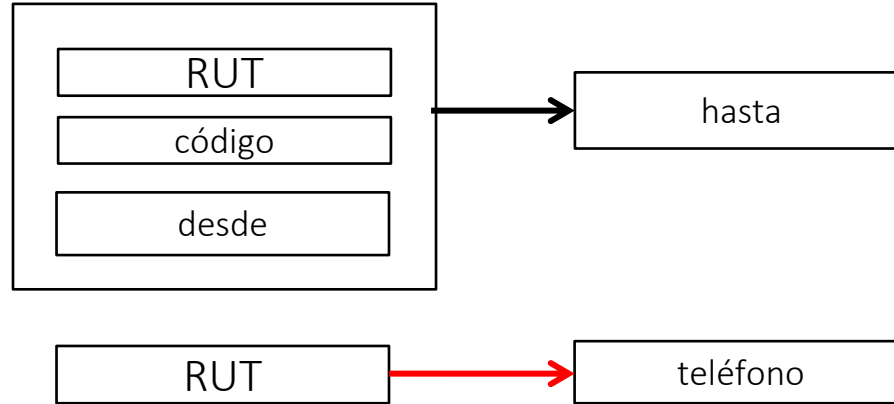
DF NO completa



<u>RUT</u>	Teléfono	<u>Código</u>	<u>desde</u>	hasta
21	9654	A1	1/3/2005	30/4/2005
21	9654	A1	5/8/2008	19/9/2008
21	9654	A2	1/3/2009	15/4/2009

- Precisamente haciendo uso de nuestro primer ejemplo, el problema que teníamos es que la dependencia del teléfono respecto del RUT es una dependencia funcional no completa ya que RUT solo es una parte de la clave primaria, teléfono solo depende de una parte de la clave primaria de la tabla en la que se encuentra.

Formas Normales



TRABAJAR

<u>RUT</u>	<u>Código</u>	<u>desde</u>	hasta
21	A1	1/3/2005	30/4/2005
21	A1	5/8/2008	19/9/2008
21	A2	1/3/2009	15/4/2009

EMPLEADO

<u>RUT</u>	teléfono
21	9654

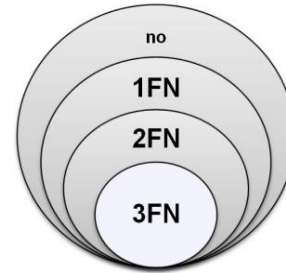


Formas Normales

- **3FN**

Una relación está en tercera forma normal (3FN) si y sólo si

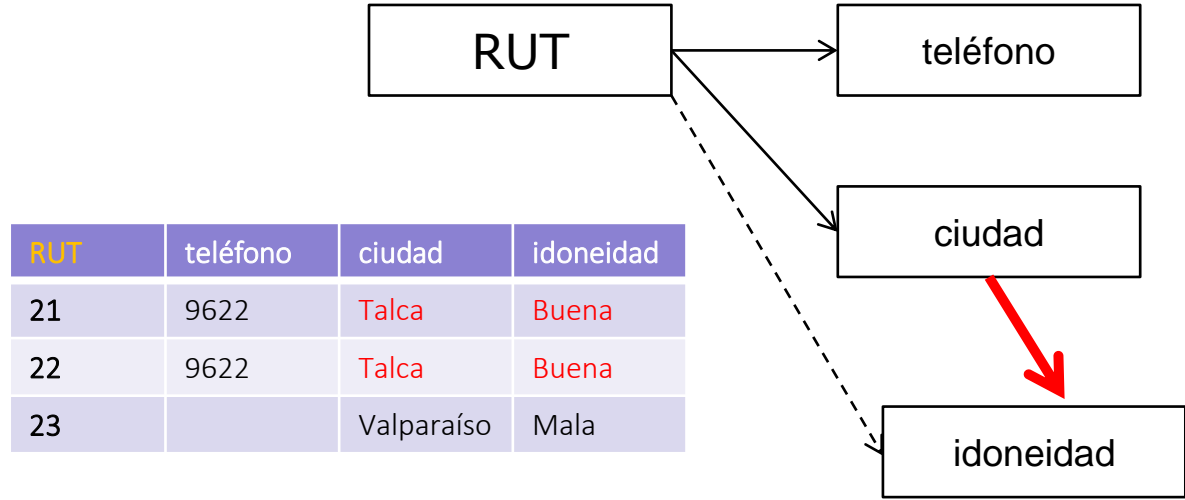
- Está en 2FN y
- todos los atributos no clave dependen de manera no transitiva de **toda** clave candidata



Formas Normales



Dependencia Transitiva

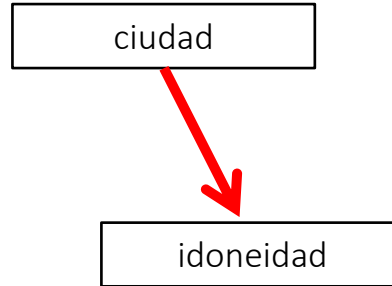
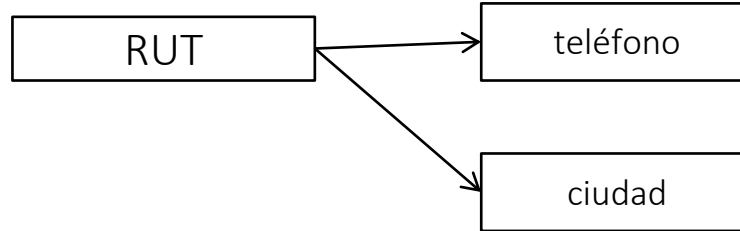


- La dependencia punteada es que está ahí pero en realidad es “el atajo” de las otras dos. A ver, si B depende de A, y C depende de B, está claro que C depende de A: ESO ES LA TRANSITIVIDAD. A veces puede estar dibujada, a veces no, pero está ahí aunque podemos ignorarla en el proceso de normalización.

Formas Normales



Normalizando



<u>RUT</u>	teléfono	ciudad
21	9622	Talca
22	9622	Talca
23		Valparaíso

<u>ciudad</u>	idoneidad
Talca	Buena
Valparaíso	Mala



Muchas gracias por tu
tiempo y participación

