

Inspira Crea Transforma

Curso de Bases de Datos

Modelado Relacional

Parte 2

MARTA SILVIA TABARES B.

Taller de Modelo Relacional

Abstracción de lo real al modelado de BDR

Taller de Modelo Relacional

Solicitud de Crédito Persona Natural



1. Información de producto a solicitar

Sede				No. Radicado			
Fecha		Valor solicitado \$		Plazo (en meses)		Solicitante <input type="checkbox"/> Codeudor <input type="checkbox"/>	
Línea de crédito		Libre inversión <input type="checkbox"/>		Vivienda <input type="checkbox"/>		Educación <input type="checkbox"/>	
		Salud <input type="checkbox"/>		Pignoración subsidio <input type="checkbox"/>		Cupo rotativo <input type="checkbox"/>	
Si su solicitud de crédito es pignoración subsidio familiar diligencie la información de los beneficiarios a pignorar:							
No. documento		Nombres y apellidos (Beneficiario)		Fecha de nacimiento		Parentesco	
1.				D D M M A A A A			
2.				D D M M A A A A			
3.				D D M M A A A A			

El taller lleva a que el estudiante pueda extraer un modelo relacional de un formulario prediseñado por una organización. En este ejercicio tomaremos dos partes del formulario.

En la figura de la parte superior se muestra la parte inicial del formulario de “Solicitud de Crédito Persona Natural de la empresa **compensar**” el cual es público y se encuentra en la url:

<http://www.compensar.com/credito/for/documents/formulariocreditopersonal.pdf>

Taller de Modelo Relacional

Solicitud de Crédito Persona Natural



1. Información de producto a solicitar

Sede				No. Radicado			
Fecha		Valor solicitado \$		Plazo (en meses)		Solicitante <input type="checkbox"/> Codeudor <input type="checkbox"/>	
Línea de crédito		Libre inversión <input type="checkbox"/>		Vivienda <input type="checkbox"/>		Educación <input type="checkbox"/>	
		Salud <input type="checkbox"/>		Pignoración subsidio <input type="checkbox"/>		Cupo rotativo <input type="checkbox"/>	
Si su solicitud de crédito es pignoración subsidio familiar diligencie la información de los beneficiarios pignorar:							
No. documento		Nombres y apellidos (Beneficiario)			Fecha de nacimiento		Parentesco
1.					D D M M A A A A		
2.					D D M M A A A A		
3.					D D M M A A A A		

Tablas (Relaciones)

1. **Producto** = {No radicado, sede, fecha, valor_solicitado, plazo, tipo_solicitante, linea_credito}
2. **Beneficiario** = {Nro documento, nombre, fecha_nacimiento, parentesco}
3. **Beneficiario_pigno** = {No radicado, nro documento}

Identificar las posibles tablas (Relaciones) que pueden verse de forma explícita en la primera parte del formulario de la Solicitud de Crédito Personal Natural.

- Partiendo del principio de que todo sustantivo es una posible Tabla, entonces se han identificado 2 tablas: “producto” (de la frase *producto a solicitar*), y “beneficiarios”.
- Para cada una de estas tablas el formulario provee un conjunto de campos que las describen, como se ve en la parte superior.
- La tabla “beneficiario_pigno” se creó para romper la relación de N:M que se puede dar porque un beneficiario puede estar relacionado con una o muchas solicitudes y una solicitud con uno o muchos beneficiarios cuando el tipo de línea de crédito se “Pignoración subsidio”. En esta tabla solo se registra información cuando la línea de crédito es de este tipo.

Taller de Modelo Relacional

Solicitud de Crédito Persona Natural



1. Información de producto a solicitar

Sede		No. Radicado	
Fecha	Valor solicitado \$	Plazo (en meses)	Solicitante <input type="checkbox"/> Codeudor <input type="checkbox"/>
Línea de crédito	Libre inversión <input type="checkbox"/> Vivienda <input type="checkbox"/> Educación <input type="checkbox"/> Salud <input type="checkbox"/>	Pignoración subsidio <input type="checkbox"/>	Cupo rotativo <input type="checkbox"/>
Si su solicitud de crédito es pignoración subsidio familiar diligencie la información de los beneficiarios pignorar:			
No. documento	Nombres y apellidos (Beneficiario)	Fecha de nacimiento	Parentesco
1.		D D M M A A A A	
2.		D D M M A A A A	
3.		D D M M A A A A	

Tablas (Relaciones)

1. **Producto** = {No radicado, sede, fecha, valor_solicitado, plazo, tipo_solicitante, linea_credito}
2. **Beneficiario** = {Nro documento, nombre, fecha_nacimiento, parentesco}
3. **Beneficiario_pigno** = {No_radicado, nro_documento}

PRODUCTO

<u>Nro_radicado</u>	sede	fecha	Valor_solicitado	plazo	Tipo_solicitante	Linea_credito
1234	Poblado	12/05/15	12.000.000	8	solicitante	LI
5678	Itagui	25/07/15	10.000.000	10	solicitante	PI
9091	City Plaza	01/02/15	5.000.000	5	solicitante	CR

BENEFICIARIOS

<u>Nro_documento</u>	nombre	Fecha_nacimiento	parentesco
12567890	Felipe Ortiz	11/03/01	hijo
23565678	Maria Gutierrez	10/23/65	esposa

Un PROCUTO pignorado puede estar relacionado con uno o muchos BENEFICIARIOS. Uno o muchos BENEFICIARIOS están relacionados con uno o muchos PRODUCTOS pignorados

<u>Nro_radicado</u>	<u>Nro_documento</u>
5678	12567890
5678	23565678

BENEFICIARIOS_PIGNO

Taller de Modelo Relacional

2. Información básica solicitante

Primer apellido		Segundo apellido		Nombres	
Tipo documento identificación C.C. <input type="checkbox"/> C.E. <input type="checkbox"/>		No. Documento identificación		Fecha de expedición D ₁ D ₂ /M ₁ M ₂ /A ₁ A ₂ A ₃ A ₄	Lugar de expedición
Fecha de nacimiento D ₁ D ₂ /M ₁ M ₂ /A ₁ A ₂ A ₃ A ₄		Ciudad de nacimiento		Departamento	País de nacimiento
Nacionalidad		Género Femenino <input type="checkbox"/> Masculino <input type="checkbox"/>		No. Personas a cargo	
Estado civil Soltero <input type="checkbox"/> Casado <input type="checkbox"/> Unión libre <input type="checkbox"/> Separado <input type="checkbox"/> Divorciado <input type="checkbox"/> Viudo <input type="checkbox"/>		Profesión		Nivel de estudios Primaria <input type="checkbox"/> Secundaria <input type="checkbox"/> Técnico o tecnológico <input type="checkbox"/> Universitario <input type="checkbox"/> Postgrado <input type="checkbox"/>	
Dirección residencia		Barrio		Departamento	Ciudad/Municipio
Comuna/Estrato		Tipo de vivienda Propia <input type="checkbox"/> Arrendada <input type="checkbox"/>		Valor arriendo \$	Teléfono celular
Teléfono residencia		Correo electrónico personal		Si la vivienda es arrendada diligencie: Nombre del arrendador	
Ciudad/Municipio		Departamento		Teléfono	

**Tabla
(Relación)**

Solicitante = {Dcto identificación, primer apellido, segundo_apellido, nombres, tipo_dcto, fecha_exp, lugar_exp, fecha_naci, ciudad_naci, departamento, pais, genero, estado_civil, nro_pers_cargo, nivel_estudios, dir_residencia, barrio, depto_resid, ciudad_resid, comuna, tipo_vivienda, valor_arriendo, tel_cel, tel_resid, e-mail, nom_arrendador, ciudad_viv, depto_viv, telefono_viv}

PRODUCTOS

Nro_radicado	sede	fecha	Valor_solicitado	plazo	Tipo_solicitante	Linea_credito	Nro_ident_solic
1234	Poblado	12/05/15	12.000.000	8	solicitante	LI	33243435
5678	Itagui	25/07/15	10.000.000	10	solicitante	PI	45464748
9091	City Plaza	01/02/15	5.000.000	5	solicitante	CR	90912837

SOLICITANTE

Dcto_identificacion	Primer_apellido	Segundo_apellido	nombre	...
45464748	Ortiz	Fernandez	Orlando	...
90912837	Velez	Zapata	Maria José	...
33243435	Alzate	Angel	Luis	...

Un SOLICITANTE puede solicitar uno o muchos PRODUCTOS y un PRODUCTO está relacionado con un SOLICITANTE

Algebra Relacional

Primer paso para aprender a consultar los datos almacenados en las tablas (Relaciones)

Algebra Relacional

- **Relación**

Conjuntos de datos ordenados en forma de filas y columnas, los cuales están relacionados por algún contexto. Las relaciones (tablas) contienen dentro de sus partes tuplas y campos.

Campo			
Campo1	Campo2	Campo3	→ Nombre del Campo o Columna
Valor 1	Valor 2	Valor 3	
Valor n	Valor n	Valor n	→ Tupla, Fila o Registro

- **Tupla**

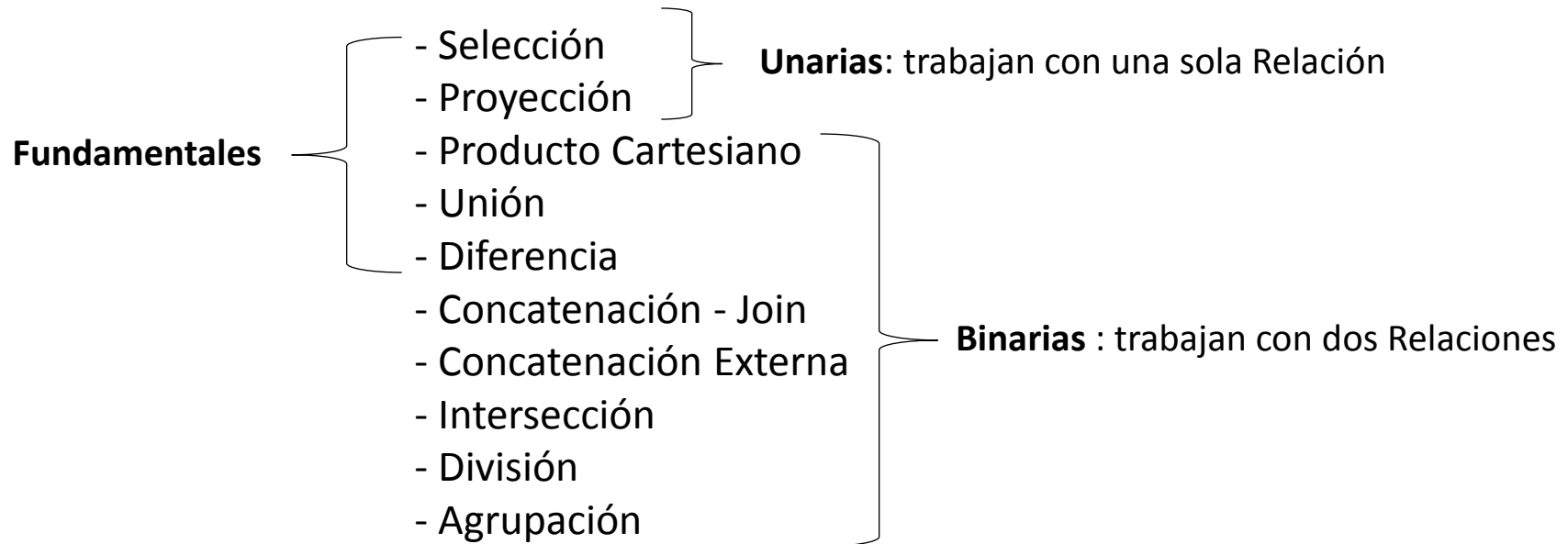
Es una parte de una relación, un conjunto de datos que entregan una información relacionada. Enfocado a las bases de datos puede llamarse “registro” ó “filas de una tabla”.

Algebra relacional

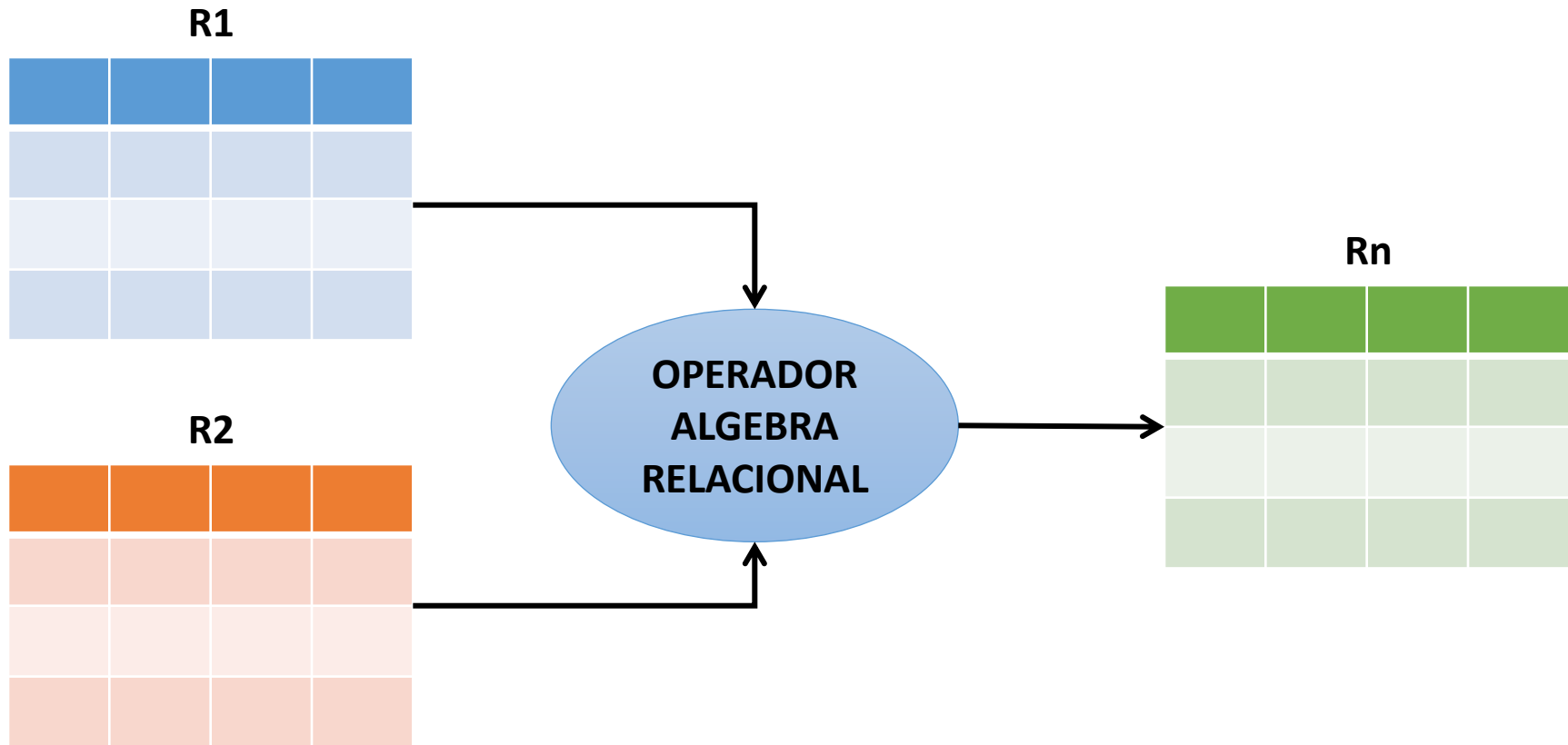
- **Aridad:** Se le denomina aridad al numero de atributos que contiene una relación.
- **Unión compatible:** Se le denomina así a la posibilidad de que dos relaciones tengan la misma aridad.
- **Dos grupos de operadores:**
 - Operadores de teoría de conjuntos: unión,
 - intersección, diferencia y producto cartesiano
 - Operadores relacionales especiales: selección,
 - proyección, reunión y división

Algebra relacional

OPERADORES ALGEBRAICOS



Algebra relacional



Algebra Relacional

OPERADORES SEGÚN SU PROCEDENCIA

de la teoría de conjuntos		propios relacionales	
\cup	Unión	σ DONDE	Selección
\cap	Intersección	π []	Proyección
\sim -	Diferencia	\bowtie ∞	Concatenación (join)
\times	Producto cartesiano	\div	División

Algebra Relacional

OPERADORES SEGÚN SU COMPLEJIDAD

básicos (o primitivas)		derivados	
\cup	Unión	\cap	Intersección
\sim -	Diferencia	\bowtie ∞	Concatenación (join)
\times	Producto cartesiano	\div	División
σ DONDE	Selección		
π []	Proyección		

Algebra relacional - EJEMPLO

Cliente

Nombre	Estado	Categoría
Ana	Hidalgo	1
Benito	Morelos	1
Carlos	Hidalgo	2
Carolina	Nayarit	3
María	Hidalgo	2
Noemí	Morelos	3
Pablo	Morelos	1
Teófilo	Chiapas	1
Tomas	Chiapas	2

ventas

Numero	ncliente	nVendedor	Monto
1	Ana	Alberto	2000
2	Benito	Crispín	3000
3	Ana	Danilo	2000
4	Carlos	Danilo	1500
5	Tomas	Crispín	1500
6	Pablo	Crispín	1300
7	Noemí	Alberto	1200
8	Noemí	Crispín	1800
9	Carlos	Danilo	2000
10	Tomas	Danilo	5000
11	Teófilo	Hernán	7000
12	Benito	Hernán	8000

Vendedor

Nombre	Edad
Alberto	30
Crispín	25
Danilo	28
Hernán	32

Fuente: <http://sistemasumma.com/2011/11/13/algebra-relacional-v-operaciones-de-conjuntos/>

Algebra relacional

- **Proyección**

Subconjunto de datos

Cliente

Nombre	Nombre	Estado	Categoría
Ana	Ana	Hidalgo	1
Benito	Benito	Morelos	1
Carlos	Carlos	Hidalgo	2
Carolina	Carolina	Nayarit	3
María	María	Hidalgo	2
Noemí	Noemí	Morelos	3
Pablo	Pablo	Morelos	1
Teófilo	Teófilo	Chiapas	1
Tomas	Tomas	Chiapas	2

$\pi_{\text{nombre}}(\text{cliente})$

Fuente: <http://sistemasumma.com/2011/11/13/algebra-relacional-v-operaciones-de-conjuntos/>

Algebra relacional

- **Selección**

Subconjunto de tuplas de una Relación que cumplen una determinada condición

$(\pi_{\text{nombre}} \sigma_{\text{nVendedor}=\text{"alberto"}} \text{ventas})$

Numero	ncliente	nVendedor	Monto
1	Ana	Alberto	2000
7	Noemí	Alberto	1200

Algebra relacional

- **Producto Cartesiano**

Concatenación de una Relación (Tabla) con todas las tuplas (registros o filas) de otra relación

EMPLEADOS		
CLAVE	NOMBRE	OFICIO
001	Juan Carlos Torres Méndez	Chofer
002	Mónica Comes Pérez	Enfermera
003	Carmen Cruz Cruz	Dentista
004	Miguel Contreras León	Ingeniero

BARRIO		
CLAVE	BARRIO	NRO.
001	Méndez	86003
002	Rovirosa	86105
003	Sabina	87100
004	Tamulte	85300

Algebra relacional

- **Producto Cartesiano**

Concatenación de una Relación (Tabla) con todas las tuplas (registros o filas) de otra relación

π Empleados.Nombre, Empleados.Oficio, Empleados.Colonia

σ Empleados.Nombre='Miguel Contreras León' \wedge Empleados.Clave = Barrio.Clave

(Empleados x Colonias)

Resultado

NOMBRE	OFICIO	BARRIO
Miguel Contreras León	Ingeniero	Tamulte

Algebra relacional

- **Producto Cartesiano**

CLAVE	NOMBRE	OFICIO	CLAVE	BARRIO	NRO.
001	Juan Carlos Torres Méndez	Chofer	001	Méndez	86003
001	Juan Carlos Torres Méndez	Chofer	002	Rovirosa	86105
001	Juan Carlos Torres Méndez	Chofer	003	Sabina	87100
001	Juan Carlos Torres Méndez	Chofer	004	Tamulte	85300
002	Mónica Comes Pérez	Enfermera	001	Méndez	86003
002	Mónica Comes Pérez	Enfermera	002	Rovirosa	86105
002	Mónica Comes Pérez	Enfermera	003	Sabina	87100
002	Mónica Comes Pérez	Enfermera	004	Tamulte	85300
003	Carmen Cruz Cruz	Dentista	001	Méndez	86003
003	Carmen Cruz Cruz	Dentista	002	Rovirosa	86105
003	Carmen Cruz Cruz	Dentista	003	Sabina	87100
003	Carmen Cruz Cruz	Dentista	004	Tamulte	85300
004	Miguel Contreras León	Ingeniero	002	Rovirosa	86105
004	Miguel Contreras León	Ingeniero	003	Sabina	87100
004	Miguel Contreras León	Ingeniero	001	Méndez	86003
004	Miguel Contreras León	Ingeniero	004	Tamulte	85300

Algebra relacional

- **Producto Cartesiano**

π Empleados.Nombre, Empleados.Oficio, Empleados.Colonia

σ Empleados.Nombre='Miguel Contreras León' \wedge Empleados.Clave = Barrio.Clave

(Empleados x Colonias)

Resultado

NOMBRE	OFICIO	BARRIO
Miguel Contreras León	Ingeniero	Tamulte

Algebra relacional

- **Concatenación -> Natural Join**

Este operador se utiliza cuando se tiene la necesidad de unir relaciones vinculando sólo las tuplas que coinciden de alguna manera.

NaturalJoin une sólo los pares de tuplas de las relaciones **R** y **S** que sean comunes. Más precisamente una tupla **r** de **R** y una tupla **s** de **S** se emparejan correctamente si y sólo si **r** y **s** coinciden en cada uno de los valores de los atributos comunes, el resultado de la vinculación es una tupla, llamada **joined tuple**.

Realizar **NaturalJoin** se obtiene una relación con los atributos de ambas relaciones y se obtiene combinando las tuplas de ambas relaciones que tengan el mismo valor en los atributos comunes.

R \bowtie **S**

Algebra relacional

- Concatenación -> Natural Join

$$R \bowtie S = \pi_{R.A1, \dots, R.An, S.A1, \dots, S.An}$$

$$(\sigma_{R.A1=S.A1 \wedge \dots \wedge R.An=S.An}(R \times S))$$

Método

1. Se realiza el producto cartesiano $R \times S$.
2. Se seleccionan aquellas filas del producto cartesiano para las que los atributos comunes tengan el mismo valor.
3. Se elimina del resultado una ocurrencia (columna) de cada uno de los atributos comunes.

Algebra relacional

- Concatenación -> Natural Join

Respuesta

Tabla Ingenieros

id	nombre	d#
123	Leon	39
234	Tomas	34
143	Josefa	25
090	Maria	34

Tabla Proyectos

d#	proyecto
39	ACU0034
34	USM7345

Ingenieros ⋈ Proyectos

id	nombre	d#	proyecto
123	Leon	39	ACU0034
234	Tomas	34	USM7345
090	Maria	34	USM7345

Fuente:
http://csrg.inf.utfsm.cl/~jfuentes/_build/html/lectures/week1/lecture3.html

Algebra relacional

- **Unión**

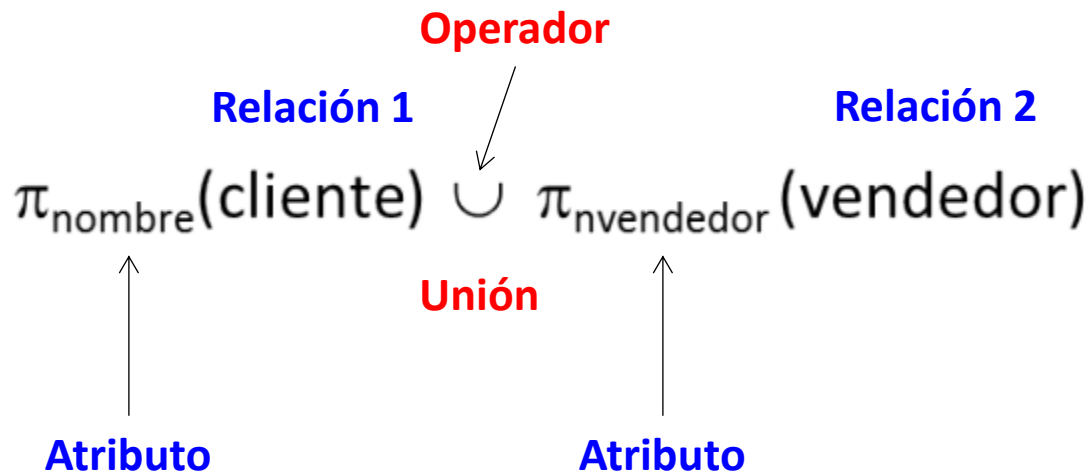
Esta operación retorna un conjunto de tuplas (tipo especial de conjunto) que estén en una o en ambas relaciones que se asocian. Para poder realizar esta operación, es necesario que las dos relaciones sean compatibles.

El resultado de esta operación entregará a lo más la cantidad de tuplas de la primera relación sumada a la cantidad de tuplas de la segunda. Los elementos repetidos aparecerán solo una vez.

$$\text{Relación1} \cup \text{Relación2}$$

Algebra relacional - EJEMPLO

Supongamos que necesitamos la lista de los nombre de todos los clientes y todos los vendedores



Nombre
Ana
Benito
Carlos
Carolina
María
Noemí
Pablo
Teófilo
Tomas
Alberto
Crispín
Danilo
Hernán

Fuente: <http://sistemasumma.com/2011/11/13/algebra-relacional-v-operaciones-de-conjuntos/>

Algebra Relacional

- **Intersección**

Esta operación obtiene como resultado la cantidad de tuplas que se encuentran en las dos relaciones. Estas relaciones deben ser compatibles para poder obtener un resultado.

$$\text{Relación1} \cap \text{Relación2}$$

Supongamos que necesitamos obtener los nombre de los clientes atendidos por ambos los vendedores “Alberto” y “Danilo”.

$$(\pi_{\text{nombre}} \sigma_{\text{nvendedor}=\text{"alberto"}} \text{ventas}) \cap (\pi_{\text{nombre}} \sigma_{\text{nvendedor}=\text{"danilo"}} \text{ventas})$$

Fuente: <http://sistemasumma.com/2011/11/13/algebra-relacional-v-operaciones-de-conjuntos/>

Algebra Relacional

- **Diferencia**

Esta operación entrega como resultado la cantidad de tuplas que se encuentren en la primera relación pero no en la segunda. Estas relaciones deben ser compatibles para poder obtener un resultado.

El resultado de esta operación, a lo más entrega el total de tuplas de la primera relación.

Relación1 - Relación2

Ejemplo: supongamos tenemos una vez mas los vendedores Alberto y Danilo, de los cuales nos interesa saber cuales fueron atendidos por Alberto pero que nunca fueron atendidos por Danilo.

$(\pi_{\text{nombre}} \sigma_{\text{nvendedor}=\text{"alberto"}} \text{ventas}) - (\pi_{\text{nombre}} \sigma_{\text{nvendedor}=\text{"danilo"}} \text{ventas})$

Restricciones de Integridad de los datos

La calidad del dato es indispensable para asegurar un buen almacenamiento y apoyar correctamente la gestión de la información en la organización.

Restricciones de Integridad

La integridad de datos se refiere a los valores reales que se almacenan y se utilizan en las estructuras de datos de un modelo relacional que ha sido definido a partir de las necesidades de manipulación de los datos.

La correcta definición de las *tablas* y sus *campos* en una *estructura* de BD, y el correcto relacionamiento de las *tablas* a través de *claves primarias* y *foráneas*, deben garantizar el control sobre todos los procesos que utilicen los datos.

Las restricciones de integridad son reglas para garantizar la calidad de los datos que son almacenados y manipulados a través del tiempo.

Restricciones de Integridad

- Restricciones sobre atributos: de dominio y de valor no nulo
- Restricciones de unicidad
- Restricción de clave primaria
- Restricción de clave ajena (integridad referencial)

Restricciones de Integridad

Restricciones sobre atributos: de dominio y de valor no nulo

- Las restricciones de dominio especifican el valor de cada atributo A .
- A debe ser un valor atómico del dominio $\text{dom}(A)$ para ese atributo, es decir es indivisible.
- Los tipos de datos asociados a los dominios por lo regular incluyen:
 - Tipos de datos numéricos estándar de los números enteros (entero-corto, entero, entero-largo) y reales (flotante, flotante de doble precisión).
 - Tipo de dato caracteres, cadenas de longitud fija y cadenas de longitud variable, así como tipos de datos de fecha, hora, marca de tiempo y dinero.
 - Otros dominios posibles se pueden ser subintervalo de valores de un tipo de datos o como un tipo de datos enumerado en el que listan explícitamente todos los valores posibles.

Restricciones de Integridad

Restricciones de Clave

- Toda tupla de una relación R debe identificarse como única a partir de un atributo o un conjunto de estos que se declare como **Clave Primaria**. Estos atributos identificarán cada tupla como única en el conjunto de tuplas de R. Esto significa que no puede haber dos tuplas que tengan la misma combinación de valores para todos sus atributos.
- Un atributo o conjunto de atributos que conformen la **Clave Primaria** **NO** puede ser nulo.

Restricciones de Integridad

Restricciones de integridad Referencial y Claves Externas

- Se especifica entre dos Relaciones y sirve para mantener la consistencia entre tuplas de dos relaciones. En términos informales, la restricción de integridad referencial establece que una tupla en una relación que haga referencia a otra relación deberá referirse a una tupla existente en cada relación.

12 Reglas de Codd

Edgar Codd en 1970 creó un sistema de manejo de datos basado en filas y columnas y cuya aplicación se hace desde la lógica proposicional y la teoría de conjuntos.

Codd define 12 reglas que identifican un modelo o sistema relacional como un todo.

- R0: Para que un sistema se denomine sistema de gestión de bases de datos relacionales, este sistema debe usar (exclusivamente) sus capacidades relacionales para gestionar la base de datos.
- R1: Toda la información en una base de datos relacional se representa explícitamente en el nivel lógico exactamente de una manera: con valores en tablas.
- R2: Para todos y cada uno de los datos (valores atómicos) de una BDR se garantiza que son accesibles a nivel lógico utilizando una combinación de nombre de tabla, valor de clave primaria y nombre de columna.

12 Reglas de Codd

- R3: Los valores nulos (que son distintos de la cadena vacía, blancos, 0, ...) se soportan en los SGBD totalmente relacionales para representar información desconocida o no aplicable de manera sistemática, independientemente del tipo de datos.
- R4: La descripción de la base de datos se representa a nivel lógico de la misma manera que los datos normales, de modo que los usuarios autorizados pueden aplicar el mismo lenguaje relacional a su consulta, igual que lo aplican a los datos normales.
- R5: Un sistema relacional debe soportar varios lenguajes y varios modos de uso de terminal (ej: rellenar formularios, etc.). Sin embargo, debe existir al menos un lenguaje cuyas sentencias sean expresables, mediante una sintaxis bien definida, como cadenas de caracteres y que sea completo, soportando: definición de datos, definición de vistas, manipulación de datos (interactiva y por programa), limitantes de integridad, limitantes de transacción (iniciar, realizar, deshacer) (Begin, commit, rollback). Además de poder tener interfaces más amigables para hacer consultas, etc. siempre debe de haber una manera de hacerlo todo de manera textual, que es tanto como decir que pueda ser incorporada en un programa tradicional.
- R6: Todas las vistas que son teóricamente actualizables se pueden actualizar por el sistema.

12 Reglas de Codd

- R7: La capacidad de manejar una relación base o derivada como un solo operando se aplica no sólo a la recuperación de los datos (consultas), si no también a la inserción, actualización y borrado de datos.
- R8: Los programas de aplicación y actividades del terminal permanecen inalterados a nivel lógico cuando quiera que se realicen cambios en las representaciones de almacenamiento o métodos de acceso.
- R9: Los programas de aplicación y actividades del terminal permanecen inalterados a nivel lógico cuandoquiera que se realicen cambios a las tablas base que preserven la información.

12 Reglas de Codd

- R10: Los limitantes de integridad específicos para una determinada base de datos relacional deben poder ser definidos en el sublenguaje de datos relacional, y almacenables en el catálogo, no en los programas de aplicación.
- R11: Una BDR tiene independencia de distribución.
- R12: Si un sistema relacional tiene un lenguaje de bajo nivel (un registro de cada vez), ese bajo nivel no puede ser usado para saltarse (subvertir) las reglas de integridad y los limitantes expresados en los lenguajes relacionales de más alto nivel (una relación (conjunto de registros) de cada vez).

Normalización de Relaciones

Divide y vencerás

Normalización

“La *normalización* es el proceso de organizar los datos de una base de datos. Se incluye la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos como para hacer que la base de datos sea más flexible al eliminar la redundancia y las dependencias incoherentes.” (Microsoft).

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN)
- Segunda Forma Normal (2FN)
- Tercera Forma Normal (3FN)
- Cuarta Forma Normal (4FN)
- Quinta Forma Normal (5FN)

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN)

- Elimine los grupos repetidos de las tablas individuales.
- Cree una tabla independiente para cada conjunto de datos relacionados.
- Identifique cada conjunto de datos relacionados con una clave principal.

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

1. PUNTO DE PARTIDA: Identificar todos los atributos que describen al solicitante (según el formulario)

2. Información básica solicitante										
Primer apellido		Segundo apellido			Nombres					
Tipo documento identificación C.C. <input type="checkbox"/> C.E. <input type="checkbox"/>		No. Documento identificación			Fecha de expedición D ₁ D ₂ /M ₁ M ₂ /A ₁ A ₂ A ₃ A ₄		Lugar de expedición		Fecha de nacimiento D ₁ D ₂ /M ₁ M ₂ /A ₁ A ₂ A ₃ A ₄	
Ciudad de nacimiento		Departamento		País de nacimiento		Nacionalidad		Género Femenino <input type="checkbox"/> Masculino <input type="checkbox"/>		
Estado civil Soltero <input type="checkbox"/> Casado <input type="checkbox"/> Unión libre <input type="checkbox"/> Separado <input type="checkbox"/> Divorciado <input type="checkbox"/> Viudo <input type="checkbox"/>								No. Personas a cargo		
Nivel de estudios Primaria <input type="checkbox"/> Secundaria <input type="checkbox"/> Técnico o tecnológico <input type="checkbox"/> Universitario <input type="checkbox"/> Postgrado <input type="checkbox"/>						Profesión				
Dirección residencia				Barrio		Departamento		Ciudad/Municipio		Comuna/Estrato
Tipo de vivienda Propia <input type="checkbox"/> Arrendada <input type="checkbox"/>		Arrendada Familiar <input type="checkbox"/> Valor arriendo \$		Teléfono celular		Teléfono residencia		Correo electrónico personal		
Si la vivienda es arrendada diligencie: Nombre del arrendador				Ciudad/Municipio		Departamento		Teléfono		

Solicitante = (primer_apell, segundo_apell, nombres, tipo_dctoid, **nro_dctoid (PK)**, fecha_exp, lugar_exp, fecha_naci, ciudad_naci, depto_naci, pais_naci, nacionalidad, genero, est_civil, nro_pers_cargo, nivel_estudios, profesion, dir_resid, barrio, dep_resid, ciudad_resid, comuna, tipo_viv, valor_arriendo, tel_cel, tel_resi, e_mail, nom_arrendador, ciudad_arrendador, dep_arrendador, tel_arrendador)

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

1. **PUNTO DE PARTIDA:** Conocer el significado de cada atributo por medio de un **diccionario de datos**.

DICcionario DE DATOS			
Nombre (etiqueta) de la columna	Tipo de Dato	NotNull	Descripción (significado) de la columna
Primer apellido	varchar(30)	X	Primer apellido del solicitante.
Segundo_apellido	varchar(30)	X	Segundo apellido del solicitante.
Nombres	varchar(50)	X	Nombres del solicitante
Tipo_dctoid	Char(2)	X	Tipo de documento del solicitante. Este puede tomar los siguientes valores cc (cédula de ciudadanía) , ce (cédula de extrangería)
Nro_dctoid	Int (Primary Key)	X	Número del documento de identidad del solicitante. Es único para cada solicitante
Fecha_exp	Date	X	Fecha de expedición del documento de identidad
Lugar_exp	Varchar(50)	X	Lugar de expedición del documento de identidad
Fecha_naci	Date	X	Fecha de nacimiento del solicitante
ciudad_naci	varchar(30)	X	Ciudad de nacimiento del solicitante
Depto_naci	varchar(30)	X	Departamento de nacimiento del solicitante

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

1. **PUNTO DE PARTIDA:** Conocer el significado de cada atributo por medio de un **diccionario de datos**.

DICCIONARIO DE DATOS			
Nombre (etiqueta) de la columna	Tipo de Dato	NotNull	Descripción (significado) de la columna
Pais_naci	varchar(30)	X	País de nacimiento del solicitante
Nacionalidad	Varchar(40)	X	Nacionalidad del participante
Genero	Char(1)	X	Género del participante. Puede tomar los siguientes valores: Femenino (F) o Masculino (M)
Est_civil	varchar(15)	X	Estado civil del participante. Puede tomar los siguientes valores: soltero, casado, unión libre, separado, divorciado, viudo.
nro_pers_cargo	Int	X	Total de personas a cargo. Puede tomar valores mayores o iguales a cero.
Nivel_de_estudios	varchar(30)	X	Nivel de estudios del solicitante. Puede tomar uno de los siguientes valores: primaria, secundaria, técnico, tecnólogo, universitario , posgrados.
Profesion	Varchar(30)	X	Profesión del solicitante

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

1. **PUNTO DE PARTIDA:** Conocer el significado de cada atributo por medio de un **diccionario de datos**.

DICCIONARIO DE DATOS			
Nombre (etiqueta) de la columna	Tipo de Dato	NotNull	Descripción (significado) de la columna
dir_resid	varchar(60)	X	Dirección de residencia del solicitante
Barrio	varchar(30)	X	Barrio en el que vive el solicitante
dep_resid	varchar(30)	X	Departamento de la residencia del solicitante
ciudad_resid,	varchar(30)	X	Ciudad de la residencia del solicitud
Comuna	varchar(30)		Comuna donde su ubica la residencia del solicitante
tipo_viv	varchar(30)	X	Tipo de vivienda del solicitante. Puede tomar los siguientes valores: propia, arrendada, familiar
valor_arriendo	varchar(30)	X	Valor del arriendo. Este valor depende si el tipo de vivienda es “arrendada”
tel_cel	Int	X	Número de teléfono celular del solicitante
Tel_resi	Int		Número de teléfono de la residencia donde vive el solicitante
e_mail	varchar(30)	X	Correo electrónico del solicitante

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

1. **PUNTO DE PARTIDA:** Conocer el significado de cada atributo por medio de un **diccionario de datos**.

DICCIONARIO DE DATOS			
Nombre (etiqueta) de la columna	Tipo de Dato	NotNull	Descripción (significado) de la columna
nom_arrendador	varchar(30)		Si la vivienda es arrendada. Nombre del arrendador.
ciudad_arrendador	varchar(30)		Si la vivienda es arrendada. Ciudad donde vive el arrendador.
dep_arrendador	varchar(30)		Si la vivienda es arrendada. Departamento donde vive el arrendador.
tel_arrendador	varchar(30)		Si la vivienda es arrendada. Teléfono del arrendador.

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

2. Identificación de columnas (atributos) que son propias de la información de la Tabla a ser normalizada, de las columnas que están relacionadas con esta, y las columnas que tienen valores constantes o conjunto de valores definidos.

- Cuáles son las **columnas propias** de una tabla?
 - Son las columnas que solo existen en la tabla, y en ninguna otra. Estas hacen que la tabla sea única en la base de datos. Describen directamente a la tabla.
- Cuáles son las **columnas que se relacionan** con una tabla?
 - Son columnas que tienen vida propia y que pueden ser complementadas por otras columnas para conformar una tabla independiente. Los datos almacenados en las nuevas tablas existen independientes de cualquier otra tabla.
 - Son columnas que pueden ser complementarias a la descripción de muchas otras tablas.

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

2. Identificación de columnas (atributos) que son propios del Solicitante y los que están relacionados con éste.

Solicitante =

(primer_apell, segundo_apell, nombres,

tipo_dctoid,

nro_dctoid (PK),

fecha_exp, lugar_exp,

fecha_naci,

ciudad_naci, depto_naci, pais_naci,

nacionalidad,

genero,

columnas propias, porque identifican o describen de manera directa al solicitante

columna relacionada, porque tiene valores definidos independiente del solicitante. Puede estar asociada a otras tablas. Los valores que toman los datos en el tiempo pueden ser variables

columna con valores propios, porque tiene valores definidos como constantes independiente del solicitante. Puede estar asociada a otras tablas

Reglas de Normalización o Formas Normales

- Primera Forma Normal (1FN) – Ejemplo

2. Identificación de columnas (atributos) que son propios del Solicitante y los que están relacionados con éste.

Solicitante =

(genero,

est_civil,

nro_pers_cargo,

nivel_estudios, profesion,

dir_resid, barrio, dep_resid, ciudad_resid, comuna, tipo_viv, valor_arriendo, tel_cel, tel_resi,
e_mail, nom_arrendador, ciudad_arrendador, dep_arrendador, tel_arrendador)

columnas propias, porque identifican o describen de manera directa al solicitante

columna relacionada, porque tiene valores definidos independiente del solicitante. Puede estar asociada a otras tablas

Reglas de Normalización o Formas Normales

- Segunda Forma Normal (2FN)
 - Cree tablas independientes para conjuntos de valores que se apliquen a varios registros.
 - Relacione estas tablas con una clave externa.

<https://support.microsoft.com/es-es/kb/283878>

Reglas de Normalización o Formas Normales

- Forma Normal (3FN)
 - Elimine los campos que no dependan de la clave.

<https://support.microsoft.com/es-es/kb/283878>

Reglas de Normalización o Formas Normales

- Cuarta y Quinta Forma Normal (4FN y 5FN)

“La cuarta forma normal, también llamada Forma normal de Boyce Codd (BCNF, Boyce Codd Normal Form), y la quinta forma normal existen, pero rara vez se consideran en un diseño real. Si no se aplican estas reglas, el diseño de la base de datos puede ser menos perfecto, pero no debería afectar a la funcionalidad.”

<https://support.microsoft.com/es-es/kb/283878>