



Dpto. Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur

ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2015

Clase 5:

**Modelo Relacional – Lenguajes de
Consulta – Algebra Relacional**

Mg. María Mercedes Vitturini
[mvitturi@uns.edu.ar]



Modelo Relacional – Definir Esquemas - Repaso

Modelo de Datos Relacional: una Base de Datos Relacional consta de múltiples *relaciones*.

- *Esquema de relación* es un conjunto finito de atributos $\{A_1, \dots, A_n\}$ tal que cada atributo A_i tiene asociado un **dominio** $\text{dom}(A_i)$.
- Una *relación* r sobre un esquema de relación R , $r(R)$ es un conjunto finito de mapeos o tuplas $\{t_1, \dots, t_p\}$ de R a D ($\text{dom}(A_1) \times \dots \times \text{dom}(A_n)$).
- Es posible definir el esquema de base de datos a partir del Modelo Entidad-Relación de forma *mecánica*.

Modelo Relacional

- Un **esquema de relación por cada entidad (fuerte o débil)**.
 - Las entidades débiles “heredan” como atributos los atributos clave de la entidad fuerte.
- Un **esquema de relación por cada relación**.
 - Si la relación no tiene atributos propios, los atributos clave de las entidades que vincula
 - Si la relación tiene atributos propios, los atributos clave de las entidades que vincula junto con los atributos de la relación.
 - Casos particulares
 - Relación is_a.
 - Relación m:1 (sin atributos).
 - Relación 1:1 (sin atributos).

Claves a partir del Modelo E-R

- **Entidades Fuertes:** La **clave primaria del conjunto entidad** es la clave primaria del esquema de relación.
- **Entidades Débiles:** La clave primaria del esquema de relación surge de la **unión de la clave primaria del conjunto entidad fuerte + el discriminador del conjunto entidad débil**.
- **Relaciones:** La unión de las claves primarias de los conjuntos entidad relacionados se transforma en:
 - Para **relaciones muchos a uno**, la clave primaria del conjunto de entidades “muchos” pasa a ser la clave primaria del esquema relación.
 - Para **relaciones uno a uno**, la clave primaria del esquema de relación puede ser la clave de cualquiera de los conjuntos de entidades.
 - Para **relaciones muchos a muchos**, la unión de las claves primarias de los conjuntos de entidades pasa a ser la clave primaria del esquema relación.
 - Si la relación tiene atributos, la clave primaria dependerá del caso.

Servicios del SMBD

1. Soporte de al menos un **Modelo de Datos**.
2. Disponibilidad de **Lenguajes de alto nivel** para administrar o manipular la base de datos
 - **Lenguaje de Manipulación de Datos** (LMD/DML).
 - **Lenguaje de Definición de Datos** (LDD/DDDL).
3. **Eficiencia** al consultar los datos almacenados.
4. Provea **manejo de transacciones** (commit y rollback).
 - Mantener **integridad** y **consistencia** de datos.
 - Provea **control de concurrencia** y capacidades para **compartir datos**.
 - Permita **recuperaciones de fallos**.
5. Brinde **seguridad** y facilidades en la **administración de datos**.

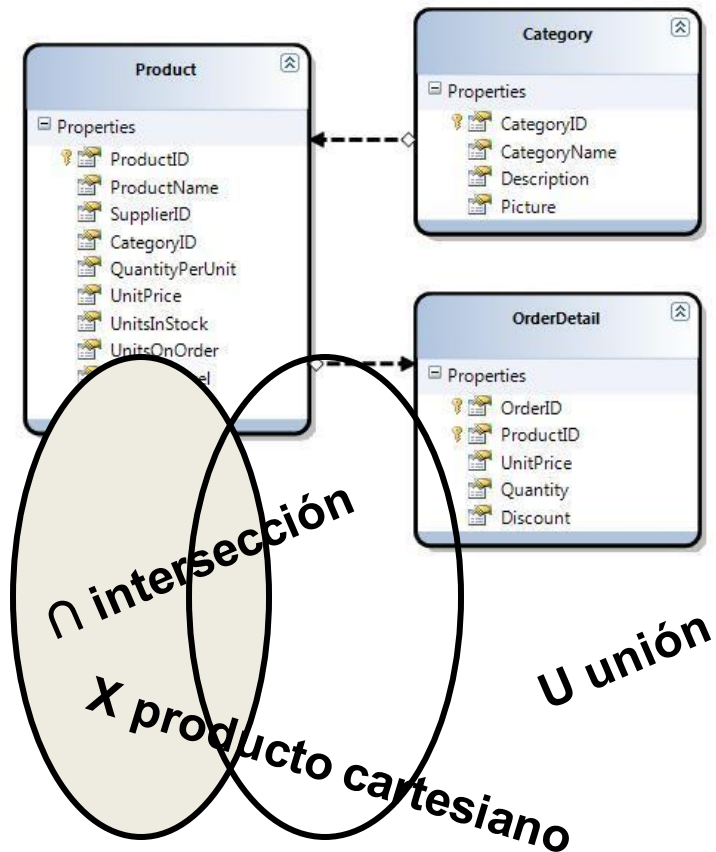
Lenguajes de Bases de Datos

- **Lenguaje de Definición de Datos (LDD/DDDL):** es el lenguaje que mediante un conjunto de sentencias sirve para especificar *el esquema de una base de datos*.
- **Lenguaje de Manipulación de Datos (LMD/DML):** es el lenguaje que nos permite *modificar y consultar la información almacenada* en una base de datos, ie. manipular instancias.

Lenguajes de Manipulación de Datos (LMD o DML)

- Categorías de lenguajes de consulta:
 - *Algebraicos o procedurales*: las consultas son expresadas aplicando operadores específicos a las relaciones.
 - Lenguaje Teórico o ‘Puro’: Algebra Relacional
 - *Lógicos o declarativos*: denominadas **cálculo relacional**, expresadas por fórmulas lógicas.
 - Lenguaje Teórico o ‘Puro’: Cálculo Relacional de Tuplas

Algebra Relacional (AR)



**Lenguaje de Consulta
Teórico orientado a
procedimientos
(procedural)**

Convenciones de notación

- Las primeras mayúsculas del alfabeto para **atributos simples** (**A, B, C, ..., L**).
- Las últimas mayúsculas para **conjuntos de atributos** (**U, V, W, X, Y, Z**).
- **R** es un **esquema de relación**. Una relación con atributos A, B y C puede notarse como **(ABC)** o **ABC** o **R(ABC)**.
- **q, r, s**: **instancias o relaciones** del esquema R.
- **A₁...A_n** se usa para representar **una tupla** {A₁, ..., A_n}.
- **XY** es abreviatura de **X ∪ Y**.
- **XA** es abreviatura de **X ∪ {A}**.

Algebra Relacional

- Lenguaje “puro” **orientado a procedimientos**. Define seis operadores básicos:
- *Operadores Unarios*
 - selección: σ
 - proyección: Π
 - Renombrar: ρ
- *Operadores Binarios*
 - unión: \cup
 - Diferencia de conjuntos: $-$
 - Producto Cartesiano: \times
- A partir de una o dos relaciones, cada operando produce una *nueva relación como resultado*.

Operador de Selección (σ)

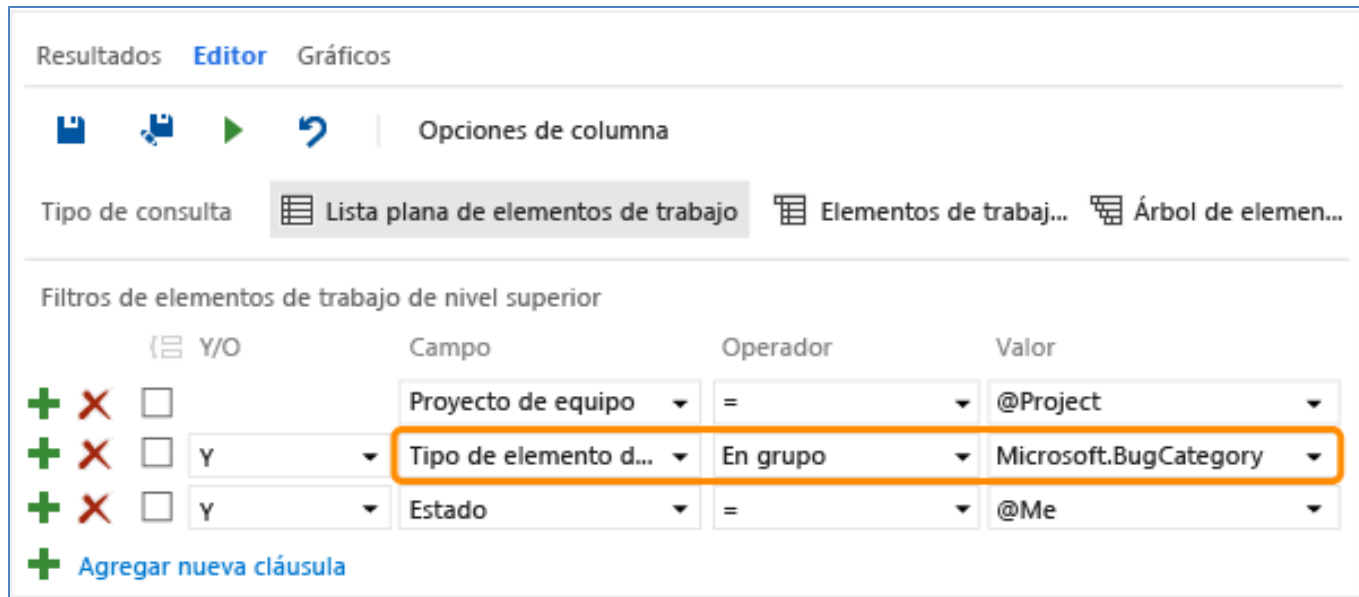
- El operador **selección** aplicado sobre una relación r , genera **una relación resultado** que contiene el *subconjunto de tuplas de r que cumplen una condición dada*.
- Se define como:

$$\sigma_p(r) = \{t \mid t \in r \wedge p(t)\}, \text{ donde:}$$

- p es una **fórmula** del cálculo proposicional, consiste de **términos** conectados por : \wedge (**and**), \vee (**or**), \neg (**not**),
- cada **término** es de la forma: $\langle \text{atributo} \rangle$ **op** $\langle \text{atributo} \rangle$ ó $\langle \text{constante} \rangle$
- y **op**: $=, \neq, >, \geq, <, \leq$

Operador Selección

- El operador selección opera como “filtro” para “recuperar” las filas que satisfacen la condición.



Resultados **Editor** Gráficos

Opciones de columna

Tipo de consulta Lista plana de elementos de trabajo Elementos de trabajaj... Árbol de elemen...

Filtros de elementos de trabajo de nivel superior

	Y/O	Campo	Operador	Valor
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>		Proyecto de equipo	=	@Project
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> Y	Y	Tipo de elemento d...	En grupo	Microsoft.BugCategory
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> Y	Y	Estado	=	@Me

Agregar nueva cláusula

Ejemplo: Operador de Selección

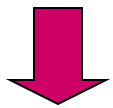
Vuelos				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
84	Chicago	JFK	15:00	17:55
109	JFK	Los Angeles	21:40	2:42
117	Atlanta	Boston	22:05	0:43
213	JFK	Boston	11:43	12:45
214	Boston	JFK	2:20	15:12

 $\sigma_{\text{Desde}=\text{"JFK"}}(\text{Vuelos})$

Vuelos desde JFK				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
109	JFK	Los Angeles	21:40	2:42
213	JFK	Boston	11:43	12:45

Operador de Selección

Vuelos				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
84	Chicago	JFK	15:00	17:55
109	JFK	Los Angeles	21:40	2:42
117	Atlanta	Boston	22:05	0:43
213	JFK	Boston	11:43	12:45
214	Boston	JFK	2:20	15:12



$\sigma_{(\text{Desde}=\text{"JFK"} \wedge \text{Hacia}=\text{"Boston"})}$ (Vuelos)

Vuelos desde JFK hacia Boston				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
213	JFK	Boston	11:43	12:45

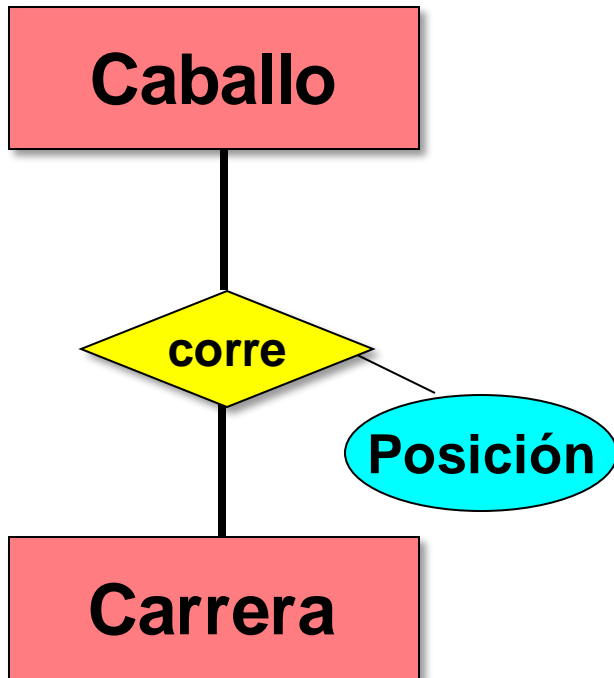
Ejemplo

El **esquema de Base de Datos**:

- Esq_Caballo (nroCaballo, nombre, peso, edad)
- Esq_Carrera (nroCarrera, nombreCarr, fecha)
- Esq_Corre (nroCaballo, nroCarrera, posición)

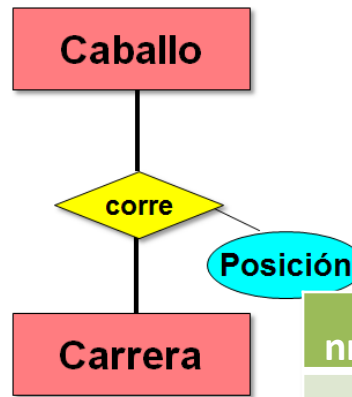
Y las **relaciones**:

- *caballo* (esq_caballo).
- *carrera* (esq_carrera).
- *corre* (esq_corre).



Ejemplo

nroCarrera	nroCaballo	posición
CC-01	CB-01	1
CC-01	CB-02	null
CC-01	CB-03	2
CC-01	CB-04	1
CC-01	CB-05	3
CC-02	CB-01	Null
CC-02	CB-03	1
CC-02	CB-04	3
CC-02	CB-05	2
CC-03	CB-01	2
CC-03	CB-02	1
CC-03	CB-03	Null
CC-03	CB-04	3
CC-03	CB-05	4
CC-04	CB-01	3
CC-04	CB-02	4
CC-04	CB-04	null
CC-04	CB-05	2



nroCarrera	nombre	fecha
CC-01	Premier	11-04-2014
CC-02	Trote	31-06-2014
CC-03	Speed	01-08-2014
CC-04	Millón	27-08-2014

carrera

nroCaballo	nombre	edad	peso
CB-01	Indio	9	280
CB-02	Orso	6	289
CB-03	Juana	2	230
CB-04	Grecia	8	245
CB-05	Chino	8	256

corre

caballo

Algunas Consultas de Selección

- ¿Qué caballos tienen tres o más años?
 - $\sigma_{\text{edad}} \geq 3$ (**caballo**).
- ¿Qué carreras se corrieron entre el mes de agosto?
 - $\sigma_{\text{fecha}} \geq \text{"01/08/2014"} \text{ and fecha} \leq \text{"31/08/2014"}$ (**carrera**).
- Caballos y carrera tal que el caballo ganó esa carrera
 - $\sigma_{\text{posición}=1}$ (**corre**).

Operador de Proyección (Π)

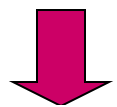
Definición – Sea r una relación sobre el esquema R y X un subconjunto de R . Se define como la "**proyección de r en X** "

$$\Pi_X(r) = \{ t(X) : t \in r \}$$

- Donde $X = (A_1, A_2 \dots A_k)$ son nombres de atributos.
- El operador **proyección** aplicado sobre una relación r , genera una relación resultado que *contiene un subconjunto de atributos (o columnas) de r* .
- El resultado se define como una relación de k columnas obtenidas de eliminar las columnas que no pertenecen a X .
- Según la teoría de conjuntos, **en $\Pi_X(r)$ se eliminan de r las tuplas (filas) repetidas.**

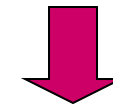
Operador de Proyección

Vuelos				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
84	Chicago	JFK	15:00	17:55
109	JFK	Los Angeles	21:40	2:42
117	Atlanta	Boston	22:05	0:43
213	JFK	Boston	11:43	12:45
214	Boston	JFK	2:20	15:12



$\Pi_{\text{Nro-Vuelo, Salida, Llegada}}(\text{vuelos})$

Nro-Vuelo	Salida	Llegada
84	15:00	17:55
109	21:40	2:42
117	22:05	0:43
213	11:43	12:45
214	2:20	15:12



$\Pi_{\text{Desde}}(\text{vuelos})$

Desde
Chicago
JFK
Atlanta
Boston

Consultas con proyección

- Nombre de fantasía de *caballos*.
 - $\Pi_{\text{nombre}}(\text{caballo})$.
- ¿En qué fechas se corrieron las carreras?
 - $\Pi_{\text{fecha}}(\text{carrera})$.

Proyección + Selección

- ¿Qué fechas del mes de agosto de 2014 se corrieron carreras?

$\Pi_{\text{fecha}}(\sigma_{\text{fecha} \geq "01/08/2014" \text{ and } \text{fecha} \leq "31/08/2014"}(\text{carrera}))$

Operador Unión (\cup)

- La unión dos relaciones r y s se define como:

$$r \cup s = \{t \mid t \in r \vee t \in s\}$$

- Para que la operación $r \cup s$ sea válida debe cumplirse:
 1. r, s deben tener la misma aridad (el mismo número de atributos).
 2. Los dominios deben ser compatibles.
- De acuerdo con la teoría de conjuntos, en la unión de relaciones *se eliminan tuplas repetidas*.

Operador Unión

<i>r</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₁	c ₁
a ₁	b ₂	c ₁
a ₂	b ₁	c ₂

<i>s</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₂	c ₁
a ₂	b ₂	c ₁
a ₂	b ₂	c ₂

<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₁	c ₁
a ₁	b ₂	c ₁
a ₂	b ₁	c ₂
a ₂	b ₂	c ₁
a ₂	b ₂	c ₂

r ∪ *s*

Ejemplo

- **ESQ-DUEÑO** (ddni, dapellido, dnombrés) y *dueños*(EDUEÑOS)
- **ESQ-JINETE** (jdni, japellido, jnombrés, jcatégoría) y *jinetes*(EJINETES)

Ejemplo:

- Nombres y apellidos de todas las personas del sistema, sean jinetes o dueños.

$\Pi_{dapellido, dnombrés}(\text{dueño}) \cup \Pi_{japellido, jnombrés}(\text{jinete})$.

Operador Diferencia

- La **diferencia** de dos conjuntos de relaciones r y s se define como:

$$r - s = \{t \mid t \in r \text{ and } t \notin s\}$$

- Para que la diferencia de conjuntos sea posible las relaciones tienen que ser **compatibles**:
 - r y s deben tener la misma aridad.
 - Los dominios de los atributos de r y s deben ser compatibles.
- **Notación**: $r - s$ ó r / s

Operador Diferencia

<i>r</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₁	c ₁
a ₁	b ₂	c ₁
a ₂	b ₁	c ₂

<i>s</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₂	c ₁
a ₂	b ₂	c ₁
a ₂	b ₂	c ₂

r \ *s* *ó* *r* - *s*

<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₁	c ₁
a ₂	b ₁	c ₂

Ejemplo

Consulta:

- Nombres y apellidos de las personas que son dueños, pero que no son jinetes

$\Pi_{\text{dapellido, dnombre}} (\text{dueño}) - \Pi_{\text{japellido, jnombre}} (\text{jinete})$

Producto Cartesiano (X)

- El **producto cartesiano** entre dos conjuntos de relaciones r y s se define como:

$$r \times s = \{tq \mid t \in r \wedge q \in s\}$$

- donde
 - Cada tupla t de r se combina con cada tupla q de s .
 - La concatenación de t y q es una tupla de $r \times s$.
- La relación resultado tiene por columnas la “*suma*” de las columnas de las relaciones r y s y filas el *producto cartesiano* de las filas de r y s .
- Si una columna A está en ambas relaciones, cada columna A del Producto Cartesiano se distingue por el “nombre de la relación” punto A : $r.A$ y $s.A$

Operador Producto Cartesiano

Vuelo				
Nro-Vuelo	Desde	Hacia	Salida	Llegada
84	Chicago	JFK	15:00	17:55
109	JFK	Los Angeles	21:40	2:42
117	Atlanta	Boston	22:05	0:43
213	JFK	Boston	11:43	12:45
214	Boston	JFK	2:20	15:12

Pasajero		
Nro-Pas	Apellido	Nombres
84	Peña	Ana Clara
109	Vitale	Matias

X



Nro-Vuelo	Desde	Hacia	Salida	Llegada	Nro-Pas	Apellido	Nombres
84	Chicago	JFK	15:00	17:55	84	Peña	Ana Clara
84	Chicago	JFK	15:00	17:55	109	Vitale	Matias
109	JFK	Los Angeles	21:40	2:42	84	Peña	Ana Clara
109	JFK	Los Angeles	21:40	2:42	109	Vitale	Matias
117	Atlanta	Boston	22:05	0:43	84	Peña	Ana Clara
117	Atlanta	Boston	22:05	0:43	109	Vitale	Matias
213	JFK	Boston	11:43	12:45	84	Peña	Ana Clara
213	JFK	Boston	11:43	12:45	109	Vitale	Matias
214	Boston	JFK	2:20	15:12	84	Peña	Ana Clara
214	Boston	JFK	2:20	15:12	109	Vitale	Matias

Pasajero
X
Vuelo

Ejemplo

DUEÑO		
ddni	dapellido	dnombre
11122233	Ron	Juan
22333444	Fuentes	María
33444555	Díez	Luis
55666777	Flores	Ana María

JINETE			
jdni	japellido	jnombre	jcategoria
11122233	Ron	Juan	C1
22333222	Ruiz	Carlos	C1
33444555	Díez	Luis	C2

- ¿Nombres y apellidos de las personas que son a la vez dueños y jinetes?

$\Pi_{dapellido, dnombre} (\sigma_{ddni=jdni} (\text{dueño x jinete}))$.

Operador Renombrar (ρ)

- El operador **renombrar** permite nombrar resultados de expresiones del álgebra relacional.

Ejemplo:

- Supongamos que $e(A_1, A_2, \dots, A_n)$ es expresión del Algebra Relacional de aridad n . La expresión:

$$\rho_{e_2(B_1, B_2, \dots, B_n)}(e)$$

- retorna el resultado de la **expresión e** bajo el nombre **e_2** , con sus **atributos nombrados** como a B_1, B_2, \dots, B_n .

Ejemplos

- Caballos.
 - $\rho_{\text{horse}}(\text{caballo}) \Rightarrow$ Renombra la relación “caballo” por el nombre “horse”
 - $\rho_{\text{horse}}(\text{number, name, age, weight})(\text{caballo}) \Rightarrow$ Renombra la relación “caballo” y sus atributos por los nombres “horse , number, name, age, weight ”

Ejercicios

- El nroCaballo de los caballos que abandonaron alguna carrera.
- El nroCaballo y nombre de los caballos que abandonaron alguna carrera.
- El nroCaballo de los caballos que no corrieron carreras.
- Nombre y edad de los caballos que en alguna carrera fueron primeros.

Operadores Adicionales

- A continuación vamos a ver cómo se definen un grupo de **operadores adicionales**
- Estos operadores *no agregan poder expresivo* al álgebra relacional, pero simplifican las consultas más frecuentes:
 - Intersección de conjuntos (\cap).
 - Join-Natural (\bowtie) y Theta-Join (\bowtie_{θ})
 - División (\div)
 - Asignación (\leftarrow)

Operador Intersección (\cap)

- Dadas dos relaciones r y s , se define r **intersección** s como:

$$r \cap s = \{ t \mid t \in r \wedge t \in s \}$$

- Como en otros operadores vistos se asume que:
 - r, s tienen la misma **aridad**.
 - Los atributos de r y s son de tipos **compatibles**.
- **Observación**: el operador intersección se obtiene a partir de la diferencia:

$$r \cap s = r - (r - s)$$

Operador Intersección

<i>r</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₁	c ₁
a ₁	b ₂	c ₁
a ₂	b ₁	c ₂

<i>s</i>		
<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₂	c ₁
a ₂	b ₂	c ₁
a ₂	b ₂	c ₂

$r \cap s$

<i>A</i>	<i>B</i>	<i>C</i>
a ₁	b ₂	c ₁

Operador Join-Natural ($|><|$)

Definición: Sean $r(R)$ y $s(S)$ dos relaciones sobre R y S respectivamente. La "*fusión (join) natural de r y s* ", notada como $r |><| s$, es el conjunto de tuplas t sobre el esquema $R \cup S$, tal que:

- Existen tuplas t_r en r y t_s en s con $t_r = t(R)$ y $t_s = t(S)$ y cada tupla de $r |><| s$ es una combinación de r y s tal que coinciden en los atributos de $R \cap S$.

$$r |><| s = \Pi_{R \cup S} (\sigma_{r.A_1=s.A_1 \wedge r.A_2=s.A_2 \wedge \dots \wedge r.A_n=s.A_n} r \times s)$$

– donde $R \cap S = \{A_1, A_2, \dots, A_n\}$.

Operador Join-Natural

<i>r</i>	
A	B
a ₁	b ₁
a ₁	b ₂
a ₂	b ₁
a ₂	b ₂
a ₃	b ₁
a ₃	b ₂

$r \bowtie s$

A	B	C
a ₁	b ₁	c ₁
a ₁	b ₁	c ₂
a ₁	b ₁	c ₃
a ₁	b ₂	c ₁
a ₂	b ₁	c ₁
a ₂	b ₁	c ₂
a ₂	b ₁	c ₃
a ₂	b ₂	c ₁
a ₃	b ₁	c ₁
a ₃	b ₁	c ₂
a ₃	b ₁	c ₃
a ₃	b ₂	c ₁

<i>s</i>	
B	C
b ₁	c ₁
b ₁	c ₂
b ₁	c ₃
b ₂	c ₁
b ₃	c ₂
b ₄	c ₃

Operador Join-Natural

<i>r</i>	
A	B
a ₁	b ₁
a ₂	b ₁

<i>s</i>	
C	D
c ₁	d ₁
c ₂	d ₁
c ₂	d ₂

$r \bowtie s$

A	B	C	D
a ₁	b ₁	c ₁	d ₁
a ₁	b ₁	c ₂	d ₁
a ₁	b ₁	c ₂	d ₂
a ₂	b ₁	c ₁	d ₁
a ₂	b ₁	c ₂	d ₁
a ₂	b ₁	c ₂	d ₂

Si las relaciones no comparten atributos es equivalente al Producto Cartesiano

Join-Natural - Definición Formal

- Sean r y s relaciones sobre los esquemas de relación R y S respectivamente.
- El **join-natural** de r y s , notado como $r \bowtie s$, se define como:

$$r \bowtie s = \Pi_{R \cup S} (\sigma_{r.A_1=s.A_1 \wedge r.A_2=s.A_2 \wedge \dots \wedge r.A_n=s.A_n} (r \times s))$$

– donde $R \cap S = \{A_1, A_2, \dots, A_n\}$.

- Si $R \cap S = \emptyset$ entonces $r \bowtie s = r \times s$.
- En muchas ocasiones se utiliza simplemente el término **join** en alusión al **join natural**.

Ejemplo de Join-Natural

Patente	Marca	Modelo
ABC-111	Corsa	2000
DEF-122	Clio	2003
RST-333	Falcon	1989
CCA-020	Megane	2002
DDE-621	Fiesta	2003

auto



Patente	cCódigo
ABC-111	c-2222
DDE-621	c-2223

venta

Patente	Marca	Modelo	cCódigo
ABC-111	Corsa	2000	c-2222
DDE-621	Fiesta	2003	c-2223

auto |><| venta

Consultas con el operador \bowtie

- Nombre de los caballos que corrieron alguna carrera.
 - $\Pi_{\text{nombre}}(\text{caballo} \bowtie \text{corre})$.
- Nombre del caballo y nombre de la carrera que corrió:
 - $\Pi_{\text{nombre}, \text{nombreCarr}}(\text{caballo} \bowtie \text{corre} \bowtie \rho_{\text{carreras}}(\text{nroCarrera}, \text{nombreCarr}, \text{fecha})(\text{carreras}))$.
- Nombre y edad de los caballos que corrieron la carrera “Millón”.
 - $\Pi_{\text{nombre}, \text{edad}}(\text{caballo} \bowtie \text{corre} \bowtie \sigma_{\text{nombreCarr}=\text{“Millón”}}(\rho_{\text{carreras}}(\text{nroCarrera}, \text{nombreCarr}, \text{fecha})(\text{carreras})))$.

Operador θ -Join

Definición: dadas dos relaciones r y s , el operador θ -Join se define como:

$$r \bowtie_{\theta} s \equiv \sigma_{\theta}(r \times s)$$

- Un operador θ -Join realiza el producto cartesiano de dos relaciones y luego resuelve la selección usando el predicado relacional θ .
- El operador θ -Join permite combinar relaciones con restricciones particulares.
 - θ es un operador relacional: $\{ >, <, \geq, \leq, =, \neq \}$.

Ejemplo

- Mostrar el nombre de los caballos mayores de 8 años que ganaron alguna carrera.

$$\Pi_{\text{nombre-caballo}}(\sigma_{\text{edad}>8 \text{ and } \text{pos}=1}(\text{caballo } |><| \text{ corre}))$$

ó

$$\Pi_{\text{Nombre-caballo}}(\text{caballo } |><| (\text{edad}>8 \wedge \text{Caballos.NroCaballo}=\text{Corre.NroCaballo} \wedge \text{pos}=1) \text{ corre})$$

Operador División (\div)

División: sean $r(R)$ y $s(S)$ relaciones con $S \subseteq R$. La operación $r \div s$ es una relación de esquema $R-S$. Donde:

- Una tupla $t \in r \div s$ si *para cada tupla t_s en s* existe una tupla t_r en r que satisface las siguientes condiciones:

$$t_r[S] = t_s[S] \quad \text{y} \quad t_r[R - S] = t$$

- **Observación:** la operación de división se define en términos de las operaciones fundamentales como:

$$r \div s = \Pi_{R-S}(r) - \Pi_{R-S}((\Pi_{R-S}(r) \times s) - \Pi_{R-S,S}(r))$$

Operador División

n Relaciones r , s :

A	B	C	D	E
α	a	α	a	1
α	a	γ	a	1
α	a	γ	b	1
β	a	γ	a	1
β	a	γ	b	3
γ	a	γ	a	1
γ	a	γ	b	1
γ	a	β	b	1

r

D	E
a	1
b	1

s

n $r \div s$

A	B	C
α	a	γ
γ	a	γ

Ejemplo: caballos que corrieron todas las carreras del mes de agosto.

- El número de las carreras que se corrieron en agosto.

$$r_1 = \Pi_{\text{NroCarrera}}(\sigma_{\text{fecha} \geq "01/08/2014 \text{ and } \text{fecha} \leq "31/08/2014"} (\text{carrera}))$$

- El nombres del caballo y el número de carrera que corrió.

$$r_2 = \Pi_{\text{Nombre, NroCarrera}} (\text{caballo } |><| \text{ corre})$$

- Ahora, los caballos que aparecen en r_2 con cada una de las carrera de r_1 .

$$\Pi_{\text{Nombre, NroCarrera}} (\text{caballo } |><| \text{ corre})$$

÷

$$\Pi_{\text{NroCarrera}}(\sigma_{\text{fecha} \geq "01/08/2014 \text{ and } \text{fecha} \leq "31/08/2014"} (\text{carrera}))$$

Operador Asignación (\leftarrow)

El operador **asignación** provee un medio conveniente para expresar consultas complejas.

Ejemplo

- $r \div s$ se podría expresar usando asignación como:

$$temp1 \leftarrow \Pi_{R-S}(r)$$

$$temp2 \leftarrow \Pi_{R-S}((temp1 \times s) - \Pi_{R-S,S}(r))$$

$$resultado \leftarrow temp1 - temp2$$

- El resultado de la derecha de \leftarrow se asigna a la **variable** a la izquierda del operador de asignación.
- La **variable** se puede usar en próximas expresiones.

Ejemplo: Recuperar los caballos que corrieron todas las carreras que se corrieron en el mes de agosto.

- Las carreras de agosto.

carrerasAgosto $\leftarrow \Pi_{\text{NroCarrera}}(\sigma_{\text{fecha} \geq "01/08/2012 \text{ and } \text{fecha} \leq "31/08/2012"}(\text{carrera}))$

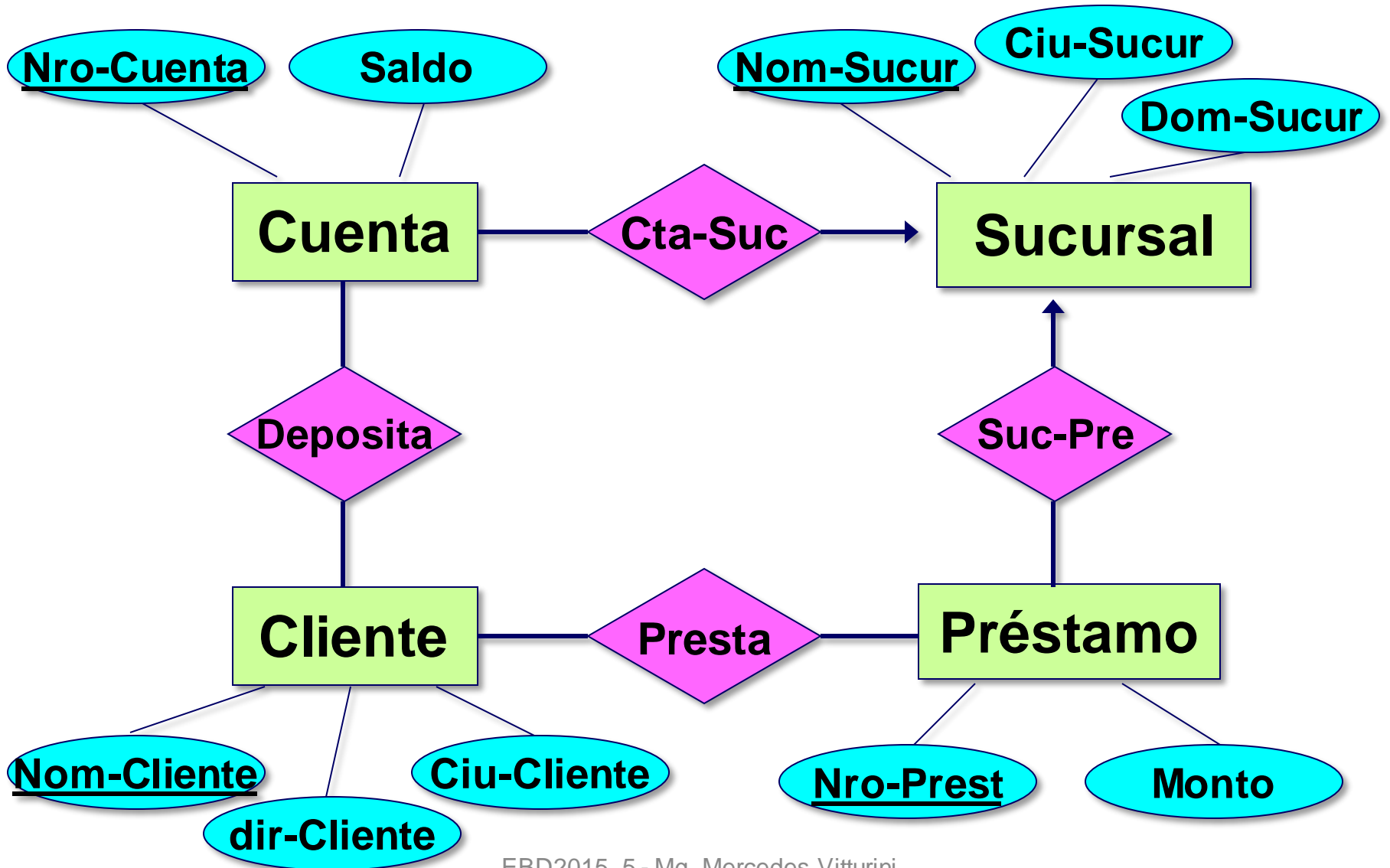
- Los nombres de caballo y nro de carrera que corrieron.

caballoCarrera $\leftarrow \Pi_{\text{Nombre}, \text{NroCarrera}}(\text{caballo} \mid \succ \mid \text{corre})$

- Ahora, los caballos que aparecen en r_2 con cada carrera de r_1 .

caballoCarrera \div **carrerasAgosto**

Diagrama E-R para un Banco



Modelo Relacional

Esquema Relacional del Modelo E/R anterior:

- Esq-Cuenta (Nro Cuenta, Nom_Sucur, Saldo).
- Esq-Cliente (Nom Cliente, Dir_Cliente, Ciu_Cliente).
- Esq-Préstamo (Nro Prest, Nom_Sucur, Monto).
- Esq-Sucursal (Nom Sucur, Dir_Sucur, Ciu_Sucur).
- Esq-Deposita (Nom Cliente, Nro Cuenta).
- Esq-Presta (Nom Cliente, Nro Prest).

Modelo Relacional

Y las relaciones:

- cuenta (Esq-Cuentas).
- cliente (Esq-Clientes).
- prestamo (Esq-Préstamos).
- sucursal (Esq-Sucursales).
- deposita (Esq-Deposita).
- presta (Esq-Presta).

Ejercicios propuestos

- Saldos de las cuentas de Pablo Garré.
- Nombres de los clientes que tienen un préstamo de la sucursal Universitario.
- Encontrar los clientes que tienen préstamos en al menos dos sucursales distintas.
- Encontrar todas las sucursales que comparten clientes con otras sucursales.
- Clientes que tienen una cuenta en todas las sucursales situadas en Tres Arroyos.

Temas de la Clase de Hoy

- Lenguajes Relacionales – Álgebra Relacional
 - Operadores:
 - Unión, Intersección, Diferencia, Selección, Proyección, Renombre, Producto Cartesiano.
 - Operadores adicionales
- **Bibliografía**
 - “Database System Concepts” – A. Silberschatz. Capítulo 2
 - “Database and Knowledge Base System” – J. Ullman. Capítulo 3.