



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

ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2015

Clase 6:

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

Mg. Walter Grandinetti



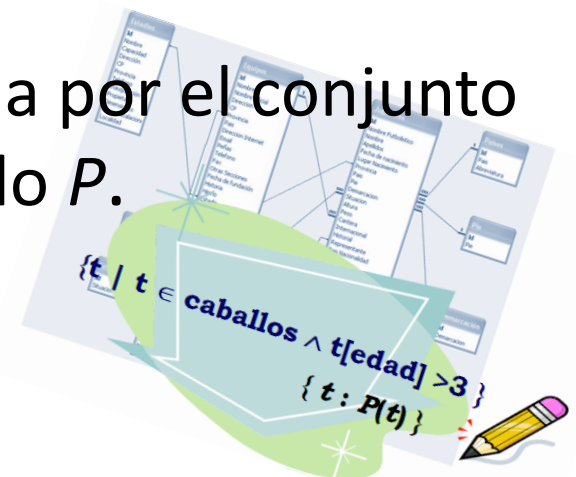
Cálculo Relacional de Tuplas

- Las **consultas en CRT se expresan en forma declarativa**, sin especificar el procedimiento efectivo para obtener la información.
- Formato general de una consulta:

$$\{ t : P(t) \}$$

Leer : *“El conjunto de tuplas t tal que satisfacen el predicado P ”.*

- El **resultado es una relación** integrada por el conjunto de tuplas t que satisfacen el predicado P .



Especificación de Conjuntos

- Método **Enumerativo**
- Método **Predicativo**: Sea S un conjunto y $P(x)$ un predicado con un parámetro x de tipo S .

$$\{ x \in S \mid P(x) \}$$

- Método **Sustitutivo**: Sea $E(x)$ una expresión.

$$\{ E(x) \mid x \in S \}$$

- Método **Híbrido**

$$\{ x \mid x \in S \wedge P(x) \}$$

Ejemplos de Conjuntos

$\{ 1, 2, 3, 4 \}$ (Enumerativo)

$\{ a, b, \{ c, d \}, \{ 3, 4 \} \}$ (Enumerativo)

$\{ t \in \text{caballo} \mid t[\text{edad}] > 8 \}$ (Predicativo)

$\{ x + 1 \mid x \in \mathbb{N} \}$ (Sustitutivo)

$\{ t \mid t \in \text{caballo} \wedge t[\text{edad}] > 8 \}$ (Híbrido)

$\{ x + 1 \mid x \in \mathbb{N} \wedge \text{mod}(x,3)=1 \}$ (Híbrido)

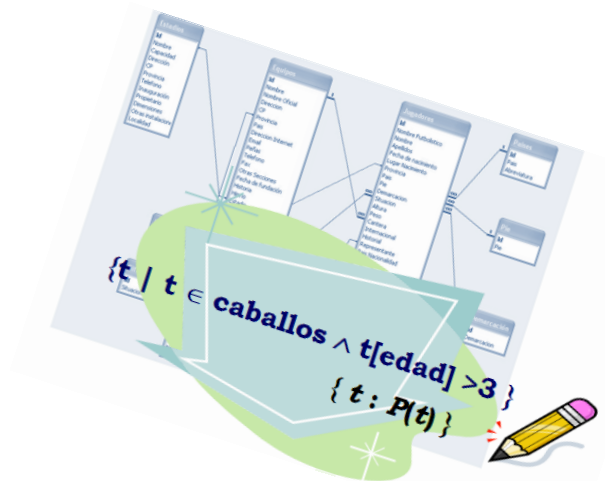
$$P1 = \{ x \in \mathbb{N} \mid \text{mod}(x,3)=1 \}$$

$$S1 = \{ x + 1 \mid x \in P1 \}$$

Cálculo Relacional de Tuplas

- Utilizaremos el método predicativo $\{ t \in S \mid P(t) \}$
Cuando se desea que las tuplas del conjunto contengan todos los atributos del esquema S.
- Utilizaremos el método híbrido $\{ t \mid t \in S \wedge P(t) \}$
Cuando se desea especificar los atributos que las tuplas han de tener, y la sintáxis será:
 $\{ t \mid P(t) \wedge t[a_1]=v_1 \wedge \dots \wedge t[a_n]=v_n \}$

- *Nota:* $\{ t \in S \mid P(t) \} = \{ t \mid t \in S \wedge P(t) \}$



CRT – Notación

- $t[A]$ denota el valor de la tupla t en el atributo A .
- $t \in r$ denota que la tupla t está en relación de pertenencia con la relación r .
- \forall : cuantificador universal “para todo”.
- \exists : cuantificador existencial “existe”.
- \wedge : conjunción lógica “y”.
- \vee : disyunción lógica “o”.
- \neg : negación lógica “no”.
- \rightarrow : implicación o entonces.

Significado de expresiones

- $\exists t \in r (Q(t))$ significa “*existe una tupla t en la relación r tal que el predicado $Q(t)$ es verdadero*”.
- $\forall t \in r (Q(t))$ significa “*el predicado $Q(t)$ es verdadero para todas las tuplas t en la relación r ”*.”.
- $P_1 \wedge P_2$ es equivalente a $\neg(\neg P_1 \vee \neg P_2)$.
- $P_1 \rightarrow P_2$ es equivalente a $\neg P_1 \vee P_2$.
- $\forall t \in r (Q(t))$ es equivalente a $\neg(\exists t \in r (\neg Q(t)))$.

Ejemplos

El **esquema de Base de Datos**:

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- **Tareas(tareaID, descrip, tipo)**
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

- *Tareas de tipo 'Desarrollo'*

$\{ t \in \text{Tareas} \mid t[\text{tipo}] = \text{'Desarrollo'} \}$

- *Proyectos de tipo 'WebApp' iniciados en el 2015*

$\{ t \in \text{Proyectos} \mid t[\text{tipo}] = \text{'WebApp'} \wedge$

$t[\text{fechaInicio}] \geq \text{'1/1/2015'} \wedge t[\text{fechaInicio}] \leq \text{'31/12/2015'} \}$

Selección

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

Proyección

- *Nombre y fecha de Ingreso del Personal*

$$\{ t \mid \exists p \in \text{Personal} (t[\text{nombre}] = p[\text{nombre}] \wedge t[\text{FechaIngreso}] = p[\text{FechaIngreso}]) \}$$

- *Nombre del Personal que ingresó a partir del '1/1/2015'*

$$\{ t \mid \exists p \in \text{Personal} (p[\text{FechaIngreso}] \geq '1/1/2015' \wedge t[\text{nombre}] = p[\text{nombre}]) \}$$

Selección +
Proyección

- **Proyectos**(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

- *Proyectos donde si el tipo es 'webApp' entonces la fecha de inicio debe ser mayor a '1/1/2010' y si el tipo es cualquier otro valor entonces no importa la fecha.*

$$\{ p \in \text{Proyectos} \mid p[\text{fechaInicio}] \geq '1/1/2010' \vee p[\text{tipo}] \neq 'WebApp' \}$$

$$\{ p \in \text{Proyectos} \mid p[\text{tipo}] = 'WebApp' \rightarrow p[\text{fechaInicio}] \geq '1/1/2010' \}$$

implicación

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

Join

- Personal asignado en *algún* proyecto

$$\{ p \in \text{Personal} \mid \exists a \in \text{Asignacion}(a[\text{persID}] = p[\text{persID}]) \}$$

- Personal asignado en *algún* proyecto que tenga tareas asignadas y registre horas en ese proyecto.

$$\{ p \in \text{Personal} \mid \exists a \in \text{Asignacion}(a[\text{persID}] = p[\text{persID}]) \wedge \exists r \in \text{RegistroHoras}(r[\text{persID}] = p[\text{persID}] \wedge a[\text{proyID}] = r[\text{proyID}]) \}$$



La tupla a no está ligada



- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)



- Personal asignado en *algún* proyecto que tenga tareas asignadas y registre horas en ese proyecto.

$$\{ p \in \text{Personal} \mid \exists a \in \text{Asignacion} (\exists r \in \text{RegistroHoras} (a[\text{persID}] = p[\text{persID}] \wedge r[\text{persID}] = p[\text{persID}] \wedge a[\text{proyID}] = r[\text{proyID}])) \}$$

- Nombre del Personal que realice tareas de limpieza

$$\{ s \mid \exists p \in \text{Personal} (\exists a \in \text{Asignacion} (\exists t \in \text{Tareas} (p[\text{persID}] = a[\text{persID}] \wedge a[\text{tareaID}] = t[\text{tareaID}] \wedge t[\text{tipo}] = \text{'Limpieza'}) \wedge p[\text{persID}] = a[\text{persID}])) \}$$

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

- *persID del personal asignado a más de una tarea en el mismo proyecto.*





$$\{p \mid \exists a_1, a_2 \in \text{Asignacion} (a_1[\text{persID}] = a_2[\text{persID}] \wedge \\ a_1[\text{proyID}] = a_2[\text{proyID}] \wedge \\ a_1[\text{tareaID}] \neq a_2[\text{tareaID}] \wedge \\ a_1[\text{persID}] = p[\text{persID}]) \}$$

Selección
múltiple sobre la
misma relación


Quantificadores Anidados


$$S = \{1,2,3,4\}$$

$$T = \{1,2\}$$

- $\forall x \in S (\exists y \in S (x=y))$ significa “Para todo (o para cada) x en S existe un y en S que verifica $x = y$ ”. 
- $\exists y \in S (\forall x \in S (x=y))$ significa “Existe un y en S para el cual todo x en S verifica que $x = y$ ”. 
- $\forall x \in S (\exists y \in T (x=y))$ significa “Para todo (o para cada) x en S existe un y en T que verifica $x = y$ ”. 
- $\forall x \in T (\exists y \in S (x=y))$ significa “Para todo (o para cada) x en T existe un y en S que verifica $x = y$ ”. 

$S = \{1,2,3,4\}$

• $\exists x \in S (\forall y \in S (x \leq y))$  $X=1$ (mínimo)

• $\forall y \in S (\exists x \in S (x \leq y))$  Para $y=1 \rightarrow X=1$
 $y=2 \rightarrow X=1$ o 2
 $y=3 \rightarrow X=1, 2$ o 3
 $y=4 \rightarrow X=1, 2, 3$ o 4

• Personal(persID, nombre, fechaIngreso)

- nombre del personal más reciente ingresado.

$\{p \mid \exists p_1 \in \text{Personal}($
 $\neg \exists p_2 \in \text{Personal}($
 $p_2[\text{fechaIngreso}] > p_1[\text{fechaIngreso}]$
 $) \wedge p_1[\text{nombre}] = p[\text{nombre}]$
 $) \}$

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

Cuantificador universal

- Personal *sin* tareas asignada.

$$\{ p \in \text{Personal} \mid \neg \exists a \in \text{Asignacion} (a[\text{persID}] = p[\text{persID}]) \}$$

$$\{ p \in \text{Personal} \mid \forall a \in \text{Asignacion} (a[\text{persID}] \neq p[\text{persID}]) \}$$

- nombre del personal y proyID de aquellas personas que estén asignadas a un proyecto pero *no hayan* registrado horas.

$$\{ t \mid \exists p \in \text{Personal} (\exists a \in \text{Asignacion} (p[\text{persID}] = a[\text{persID}] \wedge \neg \exists r \in \text{RegistroHoras} (r[\text{persID}] = a[\text{persID}] \wedge r[\text{proyID}] = a[\text{proyID}]))) \}$$

$$\{ t \mid \exists p \in \text{Personal} (\exists a \in \text{Asignacion} (p[\text{persID}] = a[\text{persID}] \wedge \forall r \in \text{RegistroHoras} (r[\text{persID}] \neq a[\text{persID}] \vee r[\text{proyID}] \neq a[\text{proyID}]))) \}$$

- **Proyectos**(proyID, clienteID, fechaInicio, tipo)
- **Personal**(persID, nombre, fechaIngreso)
- **Tareas**(tareaID, descrip, tipo)
- **Asignacion**(persID, proyID, tareaID)
- **RegistroHoras**(persID, proyID, fecha, cantHoras)

Cuantificador universal

- *Personal que haya participado de **todos** los proyectos.*

$$\{ p \in \text{Personal} \mid \forall j \in \text{Proyectos} (\exists a \in \text{Asignacion} (j[\text{proyID}] = a[\text{proyID}] \wedge a[\text{persID}] = p[\text{persID}])) \}$$

$$\{ t \mid \exists p \in \text{Personal} (\forall j \in \text{Proyectos} (\exists a \in \text{Asignacion} (j[\text{proyID}] = a[\text{proyID}] \wedge a[\text{persID}] = p[\text{persID}] \wedge p[\dots] = t[\dots]))) \}$$

- **Proyectos**(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- **RegistroHoras**(persID, proyID, fecha, cantHoras)

Cuantificador Universal

- *Proyectos que **sólo** registren más de 6 horas.*

$$\{ p \in \text{Proyecto} \mid \neg \exists r \in \text{RegistroHoras} (r[\text{proyID}] = p[\text{proyID}] \wedge r[\text{cantHoras}] \leq 6) \}$$

$$\{ p \in \text{Proyecto} \mid \forall r \in \text{RegistroHoras} (r[\text{proyID}] \neq p[\text{proyID}] \vee r[\text{cantHoras}] > 6) \}$$

$$\{ p \in \text{Proyecto} \mid \forall r \in \text{RegistroHoras} (r[\text{proyID}] = p[\text{proyID}] \rightarrow r[\text{cantHoras}] > 6) \}$$

¿Qué sucede si un proyecto no tiene horas registradas?

- Proyectos(proyID, clienteID, fechaInicio, tipo)
- Personal(persID, nombre, fechaIngreso)
- Tareas(tareaID, descrip, tipo)
- Asignacion(persID, proyID, tareaID)
- RegistroHoras(persID, proyID, fecha, cantHoras)

- *Proyld de los proyectos que **no** tengan asignadas tareas de tipo 'webApp'.*

$$\{ s \mid \exists p \in \text{Proyecto} (\forall a \in \text{Asignacion} (a[\text{proyID}] = p[\text{proyID}] \rightarrow \neg \exists t \in \text{Tareas} (t[\text{tareaID}] = a[\text{tareaID}] \wedge \text{tipo} = \text{'webapp'})) \wedge s[\text{proyID}] = p[\text{proyID}])) \}$$

$$\{ s \mid \exists p \in \text{Proyecto} (\forall a \in \text{Asignacion} (a[\text{proyID}] = p[\text{proyID}] \rightarrow \forall t \in \text{Tareas} (t[\text{tareaID}] = a[\text{tareaID}] \rightarrow \text{tipo} \neq \text{'webapp'})) \wedge s[\text{proyID}] = p[\text{proyID}])) \}$$

Definición Formal del CRT

- Dada una expresión en el CRT de la forma:

$$\{t : P(t)\}$$

- P es una **fórmula del cálculo de predicados** que puede contener *variables de tupla*.
- Una variable de tupla puede estar *libre* o *ligada*.
- Una variable está **ligada** cuando está cuantificada universalmente (\forall) o existencialmente (\exists).

Definición Formal del CRT

- Una *fórmula* del CRT *está compuesta de átomos*.
- Un **átomo** puede ser de las siguientes formas:
 - $t \in r$ donde t es una variable tupla y r una relación.
 - $t [X] \theta u [Y]$ donde:
 - t y u son variables de tupla,
 - X atributos definidos en t , Y atributos definidos en u ,
 - θ un operador relacional del conjunto $\{<, \leq, >, \geq, =, \neq\}$.
 - $t[X] \theta c$ donde:
 - t es una variable tupla, θ un operador relacional y c una constante en el dominio del atributo X .

Definición Formal del CRT

Las *fórmulas* se *construyen usando las reglas*:

- Un átomo es una fórmula.
- Si P es una fórmula entonces lo son $\neg P$ y (P) .
- Si P_1 y P_2 son fórmulas entonces lo son $P_1 \wedge P_2$, $P_1 \vee P_2$ y $P_1 \rightarrow P_2$.
- Si $P(s)$ es una fórmula y r es una relación, entonces son fórmulas:
 - $\exists s \in r (P(s))$ y
 - $\forall s \in r (P(s))$.