

Álgebra relacional

Bases de Datos

Curso 2018-2019

Jesús Correas – jcorreas@ucm.es

**Departamento de Sistemas Informáticos y Computación
Universidad Complutense de Madrid**

Bibliografía

- Bibliografía básica:
 - ▶ R. Elmasri, S.B. Navathe. **Fundamentals of Database Systems** (6a Ed). Addison-Wesley, 2010. (en español: **Fundamentos de Sistemas de Bases de Datos** (5a Ed). Addison-Wesley, 2007).
Capítulo 6 (5ª ed. en español). Capítulo 6 (6ª ed.).
- Bibliografía complementaria:
 - ▶ A. Silberschatz , H. F. Korth, S. Sudarshan. **Fundamentos de bases de datos** (5a Ed), McGraw-Hill, 2006.
Capítulo 2.

Lenguajes de consulta de BD relacionales

- Los **lenguajes de consulta** permiten obtener información de una BD relacional.
 - ▶ No permiten modificar datos de la BD, lo veremos más adelante en SQL.
- Hay dos tipos de lenguajes de consulta:
 - ▶ **Procedimentales**: El lenguaje indica al sistema una serie de operaciones que debe realizar para obtener el resultado.
 - ▶ **No procedimentales**: Se describe la información deseada sin dar un procedimiento concreto para obtener la información.
- Hay diversos **lenguajes formales** para realizar consultas sobre una BD relacional:
 - ▶ Procedimentales: **álgebra relacional**.
 - ▶ No procedimentales: **Cálculo relacional de tuplas** y **de dominios**.
- Veremos el **álgebra relacional**, que es la base de la parte de consultas del lenguaje SQL.

El álgebra relacional

- Es un **lenguaje formal** de **consulta** basado en el **álgebra de conjuntos** de la teoría matemática de conjuntos:
 - ▶ no incluye operaciones de modificación de datos.
 - ▶ Cada operación genera **una nueva relación** del MR que se puede manipular a su vez utilizando otras operaciones del álgebra.
- Las consultas se definen mediante la aplicación de una serie de **operaciones** sobre relaciones de BD:
 - ▶ Operaciones de teoría de conjuntos: **unión** \cup , **intersección** \cap , **diferencia** \setminus y **producto cartesiano** \times .
 - ▶ **Renombramiento** ρ .
 - ▶ **Selección** σ .
 - ▶ **Proyección** π .
 - ▶ **Reunión (join)** \bowtie .
 - ▶ **División** \div .
- Además se considera la **asignación** (\leftarrow) sobre relaciones temporales para dar un nombre a resultados intermedios.

Operaciones de conjuntos

- Son: **unión, intersección, diferencia y producto cartesiano**.
- Son operaciones **binarias**.
- En las tres primeras, los dos operandos deben ser instancias de esquemas de relación **con el mismo número de atributos, con igual dominio y en el mismo orden**:

$$R(A_1, \dots, A_n), S(B_1, \dots, B_n), \text{Dom}(A_i) = \text{Dom}(B_i) (\forall i \leq n)$$

- **Unión:** $R \cup S$ produce una relación con todas las tuplas que están **en R o en S** . Las tuplas duplicadas se eliminan.
- **Intersección:** $(R \cap S)(A_1, \dots, A_n)$ produce un esquema de relación con tuplas válidas aquellas que son válidas **en R y en S** .
- **Diferencia:** $(R \setminus S)(A_1, \dots, A_n)$ produce un esquema de relación con instancias válidas aquellas que son válidas **en R pero no en S** .
- El **producto cartesiano** lo veremos un poco más adelante.

Operaciones - Renombramiento

- En algunas operaciones del álgebra relacional son relevantes **los nombres de los atributos y de las relaciones**, por lo que es conveniente tener una operación para renombrarlos.
- Dado $R(A_1, \dots, A_n)$, la operación de **renombramiento** $\rho_{S(B_1, \dots, B_n)}(R)$ produce el siguiente esquema de relación:
 - ▶ El **nombre** del esquema es S .
 - ▶ **Atributos del esquema:** B_1, \dots, B_n .
 - ▶ **Instancia de la relación:** las tuplas válidas son las mismas que en R .
- **Ejemplos:** Dado el esquema $\text{EMPL}(\underline{\text{DNI}}, \text{Nombre}, \text{Ap1}, \text{Ap2}, \text{Sueldo})$:
 - ▶ **Renombrar el atributo DNI:**
 $\rho_{\text{EMPL}(\text{COD}, \text{Nombre}, \text{Ap1}, \text{Ap2}, \text{Sueldo})}(\text{EMPL})$
 - ▶ **Renombrar todos los atributos:**
 $\rho_{\text{EMPL}(\text{COD}, \text{N}, \text{A1}, \text{A2}, \text{S})}(\text{EMPL})$
 - ▶ **Renombrar el nombre del esquema a *empleados*, pero no los atributos:**
 $\rho_{\text{empleados}}(\text{EMPL})$

Operaciones - Selección

- La operación de **selección** permite seleccionar las tuplas de una relación que cumplen determinada **condición de selección** definida como una **expresión booleana**.
- Dado un esquema de relación $R(A_1, \dots, A_n)$ y una condición C sobre los atributos de R , $\sigma_C(R)$ produce un esquema de relación con las siguientes características:
 - ▶ **Atributos del esquema:** Los mismos que R .
 - ▶ **Instancia de la relación:** Las tuplas de R que hacen cierta la condición C .
- Las condiciones son expresiones lógicas de la siguiente forma:
 - $C \rightarrow \text{atributo OP atributo}$
 - $C \rightarrow \text{atributo OP constante}$
 - $C \rightarrow C \wedge C$
 - $C \rightarrow C \vee C$
 - $C \rightarrow \neg C$
 - ▶ *atributo* es un atributo de R .
 - ▶ *constante* es un valor constante.
 - ▶ **OP** es un operador de comparación:
 $\{<, >, \leq, \geq, =, \neq\}$.

Operaciones - Selección

- La operación de selección realiza una **“partición horizontal”** de la relación, quedándose con las tuplas (completas) que cumplen la condición.

- Ejemplos:** Dado el esquema $EMPL(\underline{DNI}, Nombre, Ap1, Ap2, Sueldo)$:

- ▶ Seleccionar el **empleado con DNI "27347234T"**:

$$\sigma_{DNI="27347234T"}(EMPL)$$

- ▶ Seleccionar los **empleados con salario entre 1200 y 1500 euros**:

$$\sigma_{(salario \geq 1200) \wedge (salario \leq 1500)}(EMPL)$$

- ▶ **Empleados que se apellidan García o ganan menos de 1000 euros**:

$$\sigma_{(ap1 = "García") \vee (salario < 1000)}(EMPL)$$

- ▶ La relación de entrada puede ser otra expresión del álgebra relacional. **Cuál es el significado de la siguiente expresión?**

$$\sigma_{ap1 = "García"}(\sigma_{salario < 1000}(EMPL))$$



Operaciones - Proyección

- La operación de **proyección** permite **extraer columnas** (valores de atributos) de una relación.
- Dado un esquema de relación $R(A_1, \dots, A_n)$ y un subconjunto de atributos $\{B_1, \dots, B_k\} \subseteq \{A_1, \dots, A_n\}$, la operación $\pi_{(B_1, \dots, B_k)}(R)$ produce una relación con las siguientes características:
 - ▶ **Atributos del esquema:** $\{B_1, \dots, B_k\}$.
 - ▶ **Instancia de la relación:** Las tuplas formadas por los valores de B_1, \dots, B_k en las tuplas válidas de R .
- La relación resultante **puede tener menos tuplas que la relación R** si la clave no forma parte del subconjunto de atributos.
 - ▶ El resultado de la proyección es un **conjunto** de tuplas: **no permite repetir tuplas**.
- La operación de proyección realiza una “**partición vertical**” de la relación, quedándose con las columnas seleccionadas.

Operaciones - Proyección

- **Ejemplos:** Dado el esquema $\text{EMPL}(\underline{\text{DNI}}, \text{Nombre}, \text{Ap1}, \text{Ap2}, \text{Sueldo})$:
 - ▶ Obtener **DNI y sueldo de los empleados:**

$$\pi(\text{DNI}, \text{sueldo})(\text{EMPL})$$

- ▶ **DNI de los empleados con salario superior a 1200 euros:**

$$\pi(\text{DNI})(\sigma(\text{salario} > 1200)(\text{EMPL}))$$

También se pueden utilizar **nombres de relación temporales** para los cálculos intermedios:

$$\text{Emp1} \leftarrow \sigma(\text{salario} > 1200)(\text{EMPL})$$

$$\text{Emp2} \leftarrow \pi(\text{DNI})(\text{Emp1})$$

Operaciones - Producto cartesiano

- El producto cartesiano permite **combinar los valores de dos relaciones**.
- Dados $R(A_1, \dots, A_m)$ y $S(B_1, \dots, B_n)$, el **producto cartesiano** produce un esquema $R \times S$ con:
 - ▶ **Atributos del esquema:** $m + n$ atributos: $A_1, \dots, A_m, B_1, \dots, B_n$
 - ▶ Si dos atributos A_i y B_j tienen el mismo nombre, se renombran a $R.A_i$ y $S.B_j$.
 - ▶ **Instancia de la relación:** Las tuplas válidas son las del producto cartesiando de los conjuntos de tuplas R y S .
- Se puede evitar el renombrado automático de atributos realizando una operación de renombrado antes de aplicar el producto cartesiano.
 - ▶ Si no se hace así, siempre se pueden utilizar **nombres de atributos precedidos por el nombre de la relación** en las condiciones booleanas. Por ejemplo: **EMPL.DNI = DPTO.DNI**

Operaciones - Producto cartesiano

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

- Obtención de los **nombres de los empleados que dirigen algún proyecto:**

Operaciones - Producto cartesiano

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

- Obtención de los **nombres de los empleados que dirigen algún proyecto:**

- ▶ Podemos combinar las dos tablas mediante el **producto cartesiano:**

Jefes1 \leftarrow **EMPL** \times **PROYECTO**

Así obtenemos **todas las combinaciones posibles de empleados con proyectos.**

Operaciones - Producto cartesiano

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

- Obtención de los **nombres de los empleados que dirigen algún proyecto:**

- ▶ Podemos combinar las dos tablas mediante el **producto cartesiano:**

Jefes1 \leftarrow **EMPL** \times **PROYECTO**

Así obtenemos **todas las combinaciones posibles de empleados con proyectos.**

- ▶ Después **seleccionamos** aquellas tuplas en las que el DNI del empleado coincide con el director del proyecto:

Jefes2 $\leftarrow \sigma_{(DNI=DNIDir)}(Jefes1)$

Operaciones - Producto cartesiano

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

- Obtención de los **nombres de los empleados que dirigen algún proyecto:**

- ▶ Podemos combinar las dos tablas mediante el **producto cartesiano:**

Jefes1 \leftarrow **EMPL** \times **PROYECTO**

Así obtenemos **todas las combinaciones posibles de empleados con proyectos.**

- ▶ Después **seleccionamos** aquellas tuplas en las que el DNI del empleado coincide con el director del proyecto:

Jefes2 $\leftarrow \sigma_{(\text{DNI}=\text{DNIDir})}(\text{Jefes1})$

- ▶ Por último, **proyectamos** el nombre de los directores:

Jefes $\leftarrow \pi_{(\text{Nombre}, \text{Ap1}, \text{Ap2})}(\text{Jefes2})$

- Cada director aparece una vez **aunque dirija varios proyectos.**
- Otra forma: $\pi_{(\text{Nombre}, \text{Ap1}, \text{Ap2})}(\sigma_{(\text{DNI}=\text{DNIDir})}(\text{EMPL} \times \text{PROYECTO}))$

Operaciones - Producto cartesiano

- **Otro ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención de los **nombres de los empleados que trabajan en un proyecto más de 10 horas:**

Operaciones - Producto cartesiano

- **Otro ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención de los **nombres de los empleados que trabajan en un proyecto más de 10 horas:**
 - ▶ Primero **seleccionamos** los proyectos en los que se trabaja más de 10 horas:

$\text{Proy10} \leftarrow \sigma_{\text{Horas} > 10}(\text{DEDICACION})$

Operaciones - Producto cartesiano

- **Otro ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención de los **nombres de los empleados que trabajan en un proyecto más de 10 horas:**
 - ▶ Primero **seleccionamos** los proyectos en los que se trabaja más de 10 horas:
$$\text{Proy10} \leftarrow \sigma_{\text{Horas} > 10}(\text{DEDICACION})$$
 - ▶ Después **combinamos** estos proyectos con los empleados para **seleccionar** los empleados que realizan ese trabajo:
$$\text{Empl10} \leftarrow \sigma_{(\text{DNI}=\text{DNIEmpl})}(\text{Proy10} \times \text{EMPL})$$

Operaciones - Producto cartesiano

- **Otro ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención de los **nombres de los empleados que trabajan en un proyecto más de 10 horas:**
 - ▶ Primero **seleccionamos** los proyectos en los que se trabaja más de 10 horas:
$$\text{Proy10} \leftarrow \sigma_{\text{Horas} > 10}(\text{DEDICACION})$$
 - ▶ Después **combinamos** estos proyectos con los empleados para **seleccionar** los empleados que realizan ese trabajo:
$$\text{Empl10} \leftarrow \sigma_{(\text{DNI}=\text{DNIEmpl})}(\text{Proy10} \times \text{EMPL})$$
 - ▶ Por último, **proyectamos** el nombre de los empleados:
$$\text{Resultado} \leftarrow \pi_{(\text{Nombre}, \text{Ap1}, \text{Ap2})}(\text{Empl10})$$
- O también:

$$\pi_{(\text{Nombre}, \text{Ap1}, \text{Ap2})}(\sigma_{(\text{DNI}=\text{DNIEmpl})}(\sigma_{\text{Horas} > 10}(\text{DEDICACION}) \times \text{EMPL}))$$

Operaciones - Producto cartesiano

- **Otro ejemplo más:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención del **DNI de los empleados que trabajan en al menos dos proyectos:**

Operaciones - Producto cartesiano

- **Otro ejemplo más:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención del **DNI de los empleados que trabajan en al menos dos proyectos:**

- ▶ Primero **renombramos** la relacion DEDICACION:

$$DEDR \leftarrow \rho_{DEDR}(\text{CodR}, \text{DNIR}, \text{HR})(\text{DEDICACION})$$

Operaciones - Producto cartesiano

- **Otro ejemplo más:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención del **DNI de los empleados que trabajan en al menos dos proyectos:**

- ▶ Primero **renombramos** la relacion DEDICACION:

$$\mathbf{DEDR} \leftarrow \rho_{\mathbf{DEDR}}(\mathbf{CodR}, \mathbf{DNIR}, \mathbf{HR})(\mathbf{DEDICACION})$$

- ▶ Después **combinamos** esta relacion con la relacion DEDICACION de la que procede y **seleccionamos** las tuplas que coinciden en el DNI pero no en el proyecto:

$$\mathbf{En2} \leftarrow \sigma_{(\mathbf{DNIEmpl}=\mathbf{DNIR} \wedge \mathbf{CodPr} \neq \mathbf{CodR})}(\mathbf{DEDR} \times \mathbf{DEDICACION})$$

Operaciones - Producto cartesiano

- **Otro ejemplo más:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- Obtención del **DNI de los empleados que trabajan en al menos dos proyectos:**

- ▶ Primero **renombramos** la relacion DEDICACION:

$$\text{DEDR} \leftarrow \rho_{\text{DEDR}}(\text{CodR}, \text{DNIR}, \text{HR})(\text{DEDICACION})$$

- ▶ Después **combinamos** esta relacion con la relacion DEDICACION de la que procede y **seleccionamos** las tuplas que coinciden en el DNI pero no en el proyecto:

$$\text{En2} \leftarrow \sigma_{(\text{DNIEmpl}=\text{DNIR} \wedge \text{CodPr} \neq \text{CodR})}(\text{DEDR} \times \text{DEDICACION})$$

- ▶ Por último, **proyectamos** el DNI de los empleados:

$$\text{Resultado} \leftarrow \pi_{\text{DNIEmpl}}(\text{En2})$$

- ▶ **Nota:** Este tipo de consultas se resuelve de forma diferente en SQL.

Operaciones - Reunión (Join)

- La operación de **reunión (join) condicional** corresponde a la combinación de un producto cartesiano y una selección.
- Dados $R(A_1, \dots, A_m)$ y $S(B_1, \dots, B_n)$, la **reunión condicional** de R y S se define como:

$$R \bowtie_C S = \sigma_C(R \times S)$$

- Aunque es una operación derivada de σ y \times , tiene su propio operador porque es **una de las operaciones más utilizadas**.
 - ▶ Simplifica las consultas, que pueden ser muy complejas.
- Si la condición C no se cumple para ninguna tupla, produce el conjunto vacío como resultado.
- Si alguna tupla tiene valor **NULO** en algún atributo que aparece en la condición, no se incluye en el resultado.

Operaciones - Reunión (Join)

- **Ejemplos:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNIEmpl, Horas)

- **Nombres de los empleados que dirigen algún proyecto:**

$\pi(\text{Nombre}, \text{Ap1}, \text{Ap2}) (\text{EMPL} \bowtie (\text{DNI} = \text{DNIDir}) \text{ PROYECTO})$

- **Nombres de los empleados que trabajan en un proyecto más de 10 horas:**

$\pi(\text{Nombre}, \text{Ap1}, \text{Ap2}) (\text{DEDICACION} \bowtie (\text{DNI} = \text{DNIEmpl} \wedge \text{Horas} > 10) \text{ EEMPL})$

- **Obtención del DNI de los empleados que trabajan en al menos dos proyectos:**

$\text{DEDR} \leftarrow \rho_{\text{DEDR}} (\text{CodR}, \text{DNIR}, \text{HR}) (\text{DEDICACION})$

$\text{Res} \leftarrow \pi_{\text{DNIEmpl}} (\text{DEDR} \bowtie (\text{DNIEmpl} = \text{DNIR} \wedge \text{CodPr} \neq \text{CodR}) \text{ DEDICACION})$

(Esta consulta se resuelve de forma diferente en SQL, con una función de agregación.)

Operaciones - Reunión natural (Natural Join)

- Un caso particular de **reunión condicional muy común** es aquella en la que la condición es una **conjunción de igualdades de los atributos con el mismo nombre en ambas relaciones**.

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir)

DEDICACION (CodPr, DNI, Horas)

- ▶ **Nombre de los empleados que trabajan en algún proyecto**

$\pi_{(\text{Nombre}, \text{Ap1}, \text{Ap2})}(\text{DEDICACION} \bowtie (\text{DEDICACION.DNI} = \text{EMPL.DNI}) \text{ EEMPL})$

- En el resultado se repiten los atributos incluidos en la condición con **exactamente igual valor y nombre (de distintas relaciones)**.
- Para evitar esto se utiliza la operación de **Reunión natural (natural join)**.

Operaciones - Reunión natural (Natural Join)

- Una reunión natural es como una reunión condicional en la que la condición está formada por **todos los atributos con el mismo nombre en ambos operandos**.
- Por esto, la condición **se omite** del operador \bowtie .
- Además, en el resultado **se eliminan los atributos repetidos** con el mismo valor.
- Dados $R(A_1, \dots, A_m)$ y $S(B_1, \dots, B_n)$ con atributos comunes C_1, \dots, C_j la **reunión natural** $R \bowtie S$ produce un esquema con:
 - ▶ **Atributos del esquema:** $\{A_1, \dots, A_m\} \cup \{B_1, \dots, B_n\}$ (no se repiten atributos con el mismo nombre).
 - ▶ **Instancia de la relación:** Las tuplas válidas son la combinación de tuplas válidas de R y S que coinciden en los valores de C_1, \dots, C_j .
- Si no hay atributos comunes, funciona **como el producto cartesiano**.

Operaciones - Reunión natural (Natural Join)

- **Ejemplos:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir, CodDpto)

DEPARTAMENTO (CodDpto, Nombre)

- Lista de los **proyectos con el nombre del departamento al que pertenecen:**

Operaciones - Reunión natural (Natural Join)

- **Ejemplos:** Dados los esquemas de relación:

EMPL (DNi, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir, CodDpto)

DEPARTAMENTO (CodDpto, Nombre)

- Lista de los **proyectos con el nombre del departamento al que pertenecen:**

$\pi(\text{Descripcion}, \text{Nombre})(\text{PROYECTO} \bowtie \text{DEPARTAMENTO})$

- **¿Qué devuelve la siguiente consulta?:** $\text{EMPL} \bowtie \text{DEPARTAMENTO}$

Operaciones - Reunión natural (Natural Join)

- **Ejemplos:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir, CodDpto)

DEPARTAMENTO (CodDpto, Nombre)

- Lista de los **proyectos con el nombre del departamento al que pertenecen:**

$\pi(\text{Descripcion, Nombre})(\text{PROYECTO} \bowtie \text{DEPARTAMENTO})$

- **¿Qué devuelve la siguiente consulta?:** $\text{EMPL} \bowtie \text{DEPARTAMENTO}$
- Lista de los **empleados que dirigen proyectos y los departamentos a los que pertenecen:**

Operaciones - Reunión natural (Natural Join)

- **Ejemplos:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

PROYECTO (CodPr, Descripcion, DNIDir, CodDpto)

DEPARTAMENTO (CodDpto, Nombre)

- Lista de los **proyectos con el nombre del departamento al que pertenecen:**

$\pi(\text{Descripcion}, \text{Nombre})(\text{PROYECTO} \bowtie \text{DEPARTAMENTO})$

- **¿Qué devuelve la siguiente consulta?:** $\text{EMPL} \bowtie \text{DEPARTAMENTO}$

- Lista de los **empleados que dirigen proyectos y los departamentos a los que pertenecen:**

$\text{Dirs} \leftarrow \rho_{\text{EMPL}}(\text{DNIDir}, \text{NombreDir}, \text{Ap1}, \text{Ap2}, \text{Sueldo})(\text{EMPL})$

$\text{Result} \leftarrow \pi(\text{NombreDir}, \text{Nombre})((\text{Dirs} \bowtie \text{PROYECTO}) \bowtie \text{DEPARTAMENTO})$

Extensión del álgebra relacional

- Se han propuesto múltiples operadores para **extender el álgebra relacional** para **facilitar las consultas complejas**.
- Casi todas las operaciones se pueden implementar con un conjunto básico: $\{\cup, \setminus, \times, \sigma, \pi, \rho\}$, pero las consultas resultantes son **muy complejas**.
- Por ejemplo, la **intersección** se debería implementar como:

$$R \cap S = R \setminus (R \setminus S).$$

- Por otra parte, los **operadores de reunión** $R \bowtie_{[C]} S$ solo incluyen las combinaciones de tuplas **que efectivamente coinciden en R y S** .
 - ▶ Las tuplas de R que no coinciden con ninguna de S **no aparecen en el resultado**, y viceversa.
 - ▶ Las reuniones que preservan todas las tuplas de alguno de los operandos se denominan **reuniones externas (outer joins)**.

Operaciones - Reuniones externas (Outer Joins)

- **Reunión exterior izquierda (left outer join)** $R \bowtie S$ es una reunión natural en la que se incluyen **todas las tuplas de R** combinadas con las que coincidan de S .
 - ▶ Las tuplas de R que no coincidan con ninguna de S **se rellenan con nulos en los atributos de S** .
- Del mismo modo se define la **reunión exterior derecha (right outer join)** $R \ltimes S$, que incluye todas las tuplas del operando de la derecha, combinadas con las que coincidan del operando de la izquierda.
- Por último, una **reunión exterior completa (full outer join)** $R \bowtie\!\!\!\bowtie S$ contiene todas las tuplas de las dos relaciones, combinadas con las tuplas coincidentes de la otra relación, o relleno con nulos aquellas tuplas de cada relación que no coincidan con ninguna tupla de la otra relación.

Operaciones - Reuniones externas (Outer Joins)

Ejemplo:

EMPL				
DNI	Nombre	Ap1	Ap2	Salario
37X	Juan	Sánchez	Martín	1500
24Y	Adela	García	Sanz	2300

PROYECTO		
CodPr	Descr.	DNI
4	Contabilidad	24Y
7	Marketing	55Z

Operaciones - Reuniones externas (Outer Joins)

Ejemplo:

EMPL				
DNI	Nombre	Ap1	Ap2	Salario
37X	Juan	Sánchez	Martín	1500
24Y	Adela	García	Sanz	2300

PROYECTO		
CodPr	Descr.	DNI
4	Contabilidad	24Y
7	Marketing	55Z

EMPL ⋈ PROYECTO						
DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
24Y	Adela	García	Sanz	2300	4	Contabilidad

Operaciones - Reuniones externas (Outer Joins)

Ejemplo:

EMPL					PROYECTO		
DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.	DNI
37X	Juan	Sánchez	Martín	1500	4	Contabilidad	24Y
24Y	Adela	García	Sanz	2300	7	Marketing	55Z

EMPL ⋈ PROYECTO

DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
24Y	Adela	García	Sanz	2300	4	Contabilidad

EMPL ⋈ PROYECTO

DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
37X	Juan	Sánchez	Martín	1500	NULO	NULO
24Y	Adela	García	Sanz	2300	4	Contabilidad

Operaciones - Reuniones externas (Outer Joins)

Ejemplo:

EMPL					PROYECTO		
DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.	DNI
37X	Juan	Sánchez	Martín	1500	4	Contabilidad	24Y
24Y	Adela	García	Sanz	2300	7	Marketing	55Z

EMPL ⋈ PROYECTO

DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
24Y	Adela	García	Sanz	2300	4	Contabilidad

EMPL ⋈ PROYECTO

DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
37X	Juan	Sánchez	Martín	1500	NULO	NULO
24Y	Adela	García	Sanz	2300	4	Contabilidad

EMPL ⋈ PROYECTO

DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
24Y	Adela	García	Sanz	2300	4	Contabilidad
55Z	NULO	NULO	NULO	NULO	7	Marketing

Operaciones - Reuniones externas (Outer Joins)

Ejemplo:

EMPL					PROYECTO		
DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.	DNI
37X	Juan	Sánchez	Martín	1500	4	Contabilidad	24Y
24Y	Adela	García	Sanz	2300	7	Marketing	55Z

EMPL \bowtie PROYECTO						
DNI	Nombre	Ap1	Ap2	Salario	CodPr	Descr.
37X	Juan	Sánchez	Martín	1500	NULO	NULO
24Y	Adela	García	Sanz	2300	4	Contabilidad
55Z	NULO	NULO	NULO	NULO	7	Marketing

Operaciones - División

- La operación de **división** es útil para determinadas consultas a BD.
- Permite obtener las tuplas de una relación que verifican que para algunos atributos toman **todos los valores** que aparecen en otra relación.
- Dados $R(A_1, \dots, A_m)$ y $S(B_1, \dots, B_n)$ con atributos tales que $\{B_1, \dots, B_n\} \subset \{A_1, \dots, A_m\}$, la operación $R \div S$ produce un esquema con:
 - ▶ **Atributos del esquema:** $\{C_1, \dots, C_j\} = \{A_1, \dots, A_m\} \setminus \{B_1, \dots, B_n\}$.
 - ▶ **Instancia de la relación:** Una tupla t está en $T = R \div S$ si $\{t\} \times S$ está contenido en R .
 - ▶ Informalmente, para que una tupla t esté en T , t debe aparecer combinado con **todos los valores de S** en R .
- Esta operación permite realizar consultas con **cuantificación universal**. Los SGBD no suelen implementar directamente esta operación.
- Lo vemos con un ejemplo:

Operaciones - División

- **Ejemplo:** Dados los esquemas de relación:

EMPL (DNI, Nombre, Ap1, Ap2, Sueldo)

DEDICACION (CodPr, DNI, Horas)

- Datos personales de todos los empleados que trabajan **en todos los proyectos en los que trabaja** el empleado con DNI nº 8967866R:
 - ▶ Primero se seleccionan los proyectos en los que trabaja este empleado:
 $\text{ProyEmp} \leftarrow \pi_{\text{CodPr}}(\sigma_{(\text{DNI} = '8967866R')}(\text{DEDICACION}))$
 - ▶ A continuación se seleccionan los DNI de los empleados que trabajan en cada proyecto:
 $\text{DNIProyectos} \leftarrow \pi_{(\text{CodPr}, \text{DNI})}(\text{DEDICACION})$
 - ▶ Después se obtienen los DNI de los empleados buscados mediante la operación de división:
 $\text{DNIBuscados} \leftarrow \text{DNIProyectos} \div \text{ProyEmp}$
 - ▶ Por último, se obtienen los datos personales de los empleados:
 $\text{Resultado} \leftarrow \text{DNIBuscados} \bowtie \text{EMPL}$

Tratamiento de valores nulos

- Cualquier **comparación** ($<$, $>$, \leq , \geq , $=$, \neq) con un valor **NULO** produce un resultado **desconocido**.
- Cualquier **operación lógica** con un valor **desconocido** produce un resultado **desconocido**.
- Por tanto, las condiciones lógicas pueden devolver tres valores posibles: **cierto**, **falso**, o **desconocido**.
- Las operaciones del álgebra se comportan de la siguiente forma:
- **Selección $\sigma_C(R)$** : Solo se seleccionan las tuplas de R para las que la condición C es **cierta** (si el resultado es desconocido no se incluyen).
- **Proyección, unión, intersección, diferencia**: Los valores nulos se tratan como cualquier otro valor para eliminar duplicados.

Tratamiento de valores nulos

- **Reunión condicional, reunión natural:** Se utiliza la equivalencia: $R \bowtie_C S = \sigma_C(R \times S)$.
 - ▶ Las tuplas con valor nulo en el atributo de conexión (o atributo común si es reunión natural) **no coinciden** y por tanto **no aparecen en el resultado**.
- **Reuniones externas:**
 - ▶ Funcionan de forma similar para las tuplas que **cumplen las condiciones de reunión**.
 - ▶ Para las que **no lo cumplen**, se incluyen en el resultado (dependiendo del tipo de reunión externa) rellenando con valores nulos.