Ejemplo 6.8. *Encontrar las áreas de conocimiento que tienen profesores con categoría CU o TU.*

Resultado:

$$\pi_{area}(\sigma_{(categoria=TU \vee categoria=CU)}(profesores))$$

Ejemplo 6.9. *Encontrar el nombre y la edad de aquellos alumnos que vienen de GRANADA.*

Resultado:

$$\pi_{nom\_alum,edad}(\sigma_{provicia=GRANADA}(alumnos))$$

Ejemplo 6.10. *Encontrar el DNI y el nombre de aquellos alumnos que tienen más de 25 años.*

Resultado:

$$\pi_{DNI,nom\_alum}(\sigma_{edad \geq 25}(alumnos))$$

Ejemplo 6.11. *Encontrar las provincias de las que vienen alumnos becados.*

Resultado:

$$\pi_{provincia}(\sigma_{beca=SI}(alumnos))$$

6.3. El producto cartesiano y la Θ-reunión

Vamos ahora a estudiar los operadores que permiten "conectar" tablas. De manera informal, se puede decir que, si la selección y proyección son operadores "tijera", la Θ-reunión es un operador "pegamento". Como ya comentamos, este es un operador derivado que se define a partir de otros dos más, concretamente, a partir del producto cartesiano y de la selección. Centrémonos primero en el producto cartesiano.

6.3.1. El operador producto cartesiano

El producto cartesiano es un operador binario que genera todas las tuplas que se pueden obtener combinando una tupla de la primera relación con otra tupla de la segunda relación.

Definición 6.3 (Producto Cartesiano). *Sean* $R[A_1..A_n]$ *y* $S[B_1..B_m]$ *dos relaciones cualesquiera, y sean* r *y* s *dos instancias de las mismas. El producto cartesiano* $r \times s$ *de ambas instancias es el conjunto de tuplas resultante de hacer el producto cartesiano sobre los conjuntos de tuplas de las instancias.*

Ejemplo 6.12. *Supongamos que las relaciones* $R[A, B]$ *y* $S[D]$ *tienen las instancias* r *y* s *que aparecen en la figura 6.2. El resultado del producto cartesiano también aparece en esta figura.*

A	B
a_1	b_1
a_2	b_2
a_3	b_3

$\times$

D
d_1
d_2

$=$

A	B	D
a_1	b_1	d_1
a_1	b_1	d_2
a_2	b_2	d_1
a_2	b_2	d_2
a_3	b_3	d_1
a_3	b_3	d_2

Figura 6.2: Producto cartesiano.

Las dos siguientes propiedades se deducen directamente de la definición:

Propiedad 6.1. *Sean* $R[A_1..A_n]$ *y* $S[B_1..B_m]$ *dos relaciones cualesquiera y* $W = R \times S$*. Entonces,* $W[A_1..A_n, B_1..B_m]$*. Es decir,* $esquema(W) = esquema(R) \cup esquema(S)$*.*

Propiedad 6.2. *Sean* $R[A_1..A_n]$ *y* $S[B_1..B_m]$ *dos relaciones cualesquiera, y* $W = R \times S$*. Sean* r *y* s *instancias de* R *y* S *respectivamente y* w *la correspondiente instancia de* W*. Si notamos por* $card(.)$ *la cardinalidad de una determinada instancia, se verifica* $card(w) = card(r) \times card(s)$*.*

Estas propiedades pueden comprobarse en el ejemplo anterior.

Dado que hemos introducido el primer operador binario, conviene hacer una pequeña aclaración sobre la denominación de los atributos en las expresiones del AR.

En general, cuando se realiza una sucesión de operaciones en AR, los atributos “conservan” su nombre en ellas. De hecho, en los ejemplos del apartado anterior, hemos hecho uso de esta propiedad de manera implícita, utilizando los mismos nombres de atributos para el resultado de las selecciones y para las tablas originales. No obstante, cuando se trabaja con operadores binarios que ponen en conexión dos tablas puede ocurrir que ambas tengan atributos con el mismo nombre y dichos atributos serían indistinguibles. Para evitar eso, cuando en una expresión de AR aparezcan atributos con el mismo nombre procedentes de distintas tablas, se calificaran los atributos con el nombre de la tabla a cuyo esquema pertenecen, según el formalismo *nombre-de-tabla.nombre-de-atributo*.

Ejemplo 6.13. *Si tenemos dos relaciones $R[A, B]$ y $S[A, B, C]$, según la propiedad 6.1, $R \times S$ tendrá como esquema $[R.A, R.B, S.A, S.B, S.C]$. En operaciones posteriores, podremos referirnos a R.A y R.B, por ejemplo, sin ningún problema. Para nombrar al atributo C de la relación S podemos usar S.C o C simplemente, ya que no hay confusión posible.*

6.3.2. La Θ-reunión

Como hemos comentado antes, el operador producto cartesiano sirve para conectar dos o más tablas, pero es un “operador mecánico” en el sentido de que conecta cada tupla con todas las demás sin tener en cuenta las conexiones semánticas que existen entre las tuplas. En realidad, el producto cartesiano, considerado individualmente, no sirve para mucho. Su objetivo es producir tablas sobre las que se pueda seleccionar y proyectar, para obtener la información que se desea. El siguiente ejemplo ilustra esta idea:

Ejemplo 6.14. *Consideremos nuestra base de datos de ejemplo y supongamos que deseamos saber, para cada departamento, el nombre de su director.*

Si conectamos la tabla Departamentos con la tabla Profesores mediante el producto cartesiano y hacemos $profesores \times departamentos$, obtenemos la tabla 6.11.

Dicha tabla incluye 50 tuplas y, realmente, no nos da lo que buscamos, ya que ha conectado todas la tuplas de profesores con todas las tuplas de departamentos. Para obtener el nombre de los directores de departamento, necesitamos identificar el atributo director de la tabla Departamentos, que corresponde al NRP del director del Departamento, con el atributo NRP de la tabla profesores. Sólo nos interesa obtener

NRP	NOM_PROF	CATG.	AREA.	COD_DEP	COD_DEP	NOM_DEP	DIRECTOR
2428456	Juan Sanchez Perez	AS	COMPUT	CCIA	CCIA	Ciencias de la Computacion	24283256
24283256	Antonia Perez Rodriguez	CU	COMPUT	CCIA	CCIA	Ciencias de la Computacion	24283256
242256	Luis Perez Perez	TE	LENGUA	LSI	CCIA	Ciencias de la Computacion	24283256
84256	Carmen Perez Sanchez	TU	LENGUA	LSI	CCIA	Ciencias de la Computacion	24283256
324256	David Perez Jimenez	CU	ARQUIT	ATC	CCIA	Ciencias de la Computacion	24283256
24256	Maria Lopez Ruiz	TU	ARQUIT	ATC	CCIA	Ciencias de la Computacion	24283256
2842560	Jose Alvarez Perez	CE	ELECTR	ELEC	CCIA	Ciencias de la Computacion	24283256
842560	Adela Perez Sanchez	AS	ELECTR	ELEC	CCIA	Ciencias de la Computacion	24283256
84560	Luis Martinez Perez	AS	TSEÑAL	TESE	CCIA	Ciencias de la Computacion	24283256
242560	Maria Gomez Sanchez	CU	TSEÑAL	TESE	CCIA	Ciencias de la Computacion	24283256
2428456	Juan Sanchez Perez	AS	COMPUT	CCIA	LSI	Lenguajes y Sistemas	84256
...	...	...	...	...	...	...	...
2428456	Juan Sanchez Perez	AS	COMPUT	CCIA	ATC	Arquitectura de Computadores	324256
...	...	...	...	...	...	...	...
2428456	Juan Sanchez Perez	AS	COMPUT	CCIA	ELEC	Electronica	2842560
...	...	...	...	...	...	...	...
2428456	Juan Sanchez Perez	AS	COMPUT	CCIA	TESE	Teoria de la Señal	84560
24283256	Antonia Perez Rodriguez	CU	COMPUT	CCIA	TESE	Teoria de la Señal	84560
242256	Luis Perez Perez	TE	LENGUA	LSI	TESE	Teoria de la Señal	84560
84256	Carmen Perez Sanchez	TU	LENGUA	LSI	TESE	Teoria de la Señal	84560
324256	David Perez Jimenez	CU	ARQUIT	ATC	TESE	Teoria de la Señal	84560
24256	Maria Lopez Ruiz	TU	ARQUIT	ATC	TESE	Teoria de la Señal	84560
2842560	Jose Alvarez Perez	CE	ELECTR	ELEC	TESE	Teoria de la Señal	84560
842560	Adela Perez Sanchez	AS	ELECTR	ELEC	TESE	Teoria de la Señal	84560
84560	Luis Martinez Perez	AS	TSEÑAL	TESE	TESE	Teoria de la Señal	84560
242560	Maria Gomez Sanchez	CU	TSEÑAL	TESE	TESE	Teoria de la Señal	84560

Tabla 6.11: Resultado de $profesores \times departamentos$.

NRP	NOM_PROF	CATG.	AREA.	COD_DEP	COD_DEP	NOM_DEP	DIRECTOR
24283256	Antonia Perez Rodriguez	CU	COMPUT	CCIA	CCIA	Ciencias de la Computacion	24283256
84256	Carmen Perez Sanchez	TU	LENGUA	LSI	LSI	Lenguajes y Sistemas	84256
324256	David Perez Jimenez	CU	ARQUIT	ATC	ATC	Arquitectura de Computadores	324256
2842560	Jose Alvarez Perez	CE	ELECTR	ELEC	ELEC	Electronica	2842560
84560	Luis Martinez Perez	AS	TSEÑAL	TESE	TESE	Teoria de la Señal	84560

Tabla 6.12: Resultado de $\sigma_{director=NRP}(profesores \times departamentos)$.

esas tuplas, que serán las que debemos seleccionar a partir del producto cartesiano anterior. Es decir, debemos calcular:

$$\sigma_{director=NRP}(profesores \times departamentos)$$

El resultado de la expresión anterior aparece en la tabla 6.12.

Esta tabla nos proporciona las tuplas donde se muestra la conexión de la tabla profesores con departamentos a través de la relación de dirección. No obstante, no es aún la respuesta correcta a nuestra consulta, ya que ofrece información redundante (¡incluso tiene dos columnas repetidas!). Para obtener un resultado perfecto, debemos proyectar sobre los atributos que nos interesan. La expresión final que nos proporciona los nombres de los departamentos junto con los de sus directores es:

$$\pi_{nom\_prof,nom\_dep}(\sigma_{director=NRP}(profesores \times departamentos))$$

NOM_PROF	NOM_DEP
Antonia Perez Rodriguez	Ciencias de la Computacion
Carmen Perez Sanchez	Lenguajes y Sistemas
David Perez Jimenez	Arquitectura de Computadores
Jose Alvarez Perez	Electronica
Luis Martinez Perez	Teoria de la Señal

Tabla 6.13: Resultado final de la consulta.

El resultado de la consulta aparece en la tabla 6.13.

Vemos, pues, que se hace necesario combinar el producto cartesiano con una selección para que esta operación tenga sentido. Esto conduce a la operación Θ-reunión (Θ-join) que pasamos a definir formalmente:

Definición 6.4. *Sean $R[A_1..A_n]$, y $S[B_1..B_m]$ dos relaciones cualesquiera, Θ una propiedad que implica a atributos de ambas relaciones y r y s dos instancias de las misma, definimos:*

$$r \bowtie_{\Theta} s = \sigma_{\Theta}(r \times s)$$

Habitualmente, esta operación se expresa como selección y producto cartesiano, no como $\bowtie_{\Theta}$.

La Θ-reunión es la operación que permite verderamente "pegar" dos tablas y restablecer las conexiones semánticas existentes entre ellas. La secuencia para la aplicación de los operadores que intervienen en la Θ-reunión es siempre la misma:

1. Determinar las tablas que intervienen en la consulta, si es necesario con una selección previa.
2. Determinar los atributos comunes a ambas tablas que establecen la conexión semántica entre ellas.
3. Formular la selección y el producto cartesiano.
4. Proyectar para obtener los atributos adecuados.

Los siguientes ejemplos ilustran este proceso de reunión:

Ejemplo 6.15. *Obtener para cada profesor, su NRP, su nombre y el nombre del departamento al que pertenece.*

El proceso de resolución es:

1. Las tablas implicadas son profesores y departamentos.
2. Los atributos comunes a estas tablas, que establecen la conexión de pertenencia, son los códigos de departamento (COD_DEP).
3. La tabla completa sería:

$$\sigma_{departamentos.cod\_dep=profesores.cod\_dep}(departamentos \times profesores)$$

4. Proyectando sobre los atributos deseados, obtenemos la solución a la consulta:

$$\pi_{NRP,nom\_prof,nom\_dep}(\sigma_{departamentos.cod\_dep=profesores.cod\_dep}$$
$$(departamentos \times profesores))$$

Ejemplo 6.16. *Obtener el DNI y el nombre de aquellos alumnos matriculados de la asignatura de código BDI que son becarios.*

1. Tablas implicadas: $\sigma_{cod\_asig='BDI'}(matriculas)$ para considerar las matrículas de la asignatura de código 'BDI' y $\sigma_{beca='SI'}(alumnos)$ para considerar sólo los alumnos becarios.
2. Atributos relevantes: DNI de la tabla MATRICULA y DNI de la tabla ALUMNOS.
3. La tabla completa es:

$$\sigma_{alumnos.DNI=matriculas.DNI}(\sigma_{beca='SI'}(alumnos)\times$$
$$\times\sigma_{cod\_asig='BDI'}(matriculas))$$

4. La solución a la consulta es:

$$\pi_{alumnos.DNI,nom\_alum}(\sigma_{alumnos.DNI=matriculas.DNI}$$
$$(\sigma_{beca='SI'}(alumnos) \times \sigma_{cod\_asig='BDI'}(matriculas)))$$

En este último ejemplo podemos observar dos aspectos interesantes. El primero es la utilización de dos atributos de dos tablas distintas con el mismo nombre y cómo se pueden distinguir utilizando los nombres de las tablas. El segundo es el hecho de que pueden aparecer varias selecciones en una expresión. Cuando esto ocurre, la expresión puede simplificarse utilizando una única propiedad compuesta. Concretamente, en nuestro ejemplo, podemos sustituir la expresión de la consulta por:

$$\pi_{alumnos.DNI,nom\_alum}(\sigma_{(alumnos.DNI=matriculas.DNI)\wedge \wedge(beca='SI')\wedge(cod\_asig='BDI')}(alumnos \times matriculas))$$

Esta forma es totalmente equivalente a la anterior desde el punto de vista del AR, que es un lenguaje formal. No ocurre así desde el punto de vista de la resolución de la consulta por medio de un sistema de gestión de bases de datos cualquiera. Hay que tener en cuenta que el producto cartesiano genera siempre tablas muy grandes y, por tanto, cuantas menos tuplas tengan las tablas que intervienen en él, más eficientemente se resolverá la consulta. En este sentido, la primera forma de solucionar esta consulta es mucho más eficiente que la segunda, ya que hacemos el producto cartesiano de tablas previamente seleccionadas y, por tanto, más pequeñas.

A continuación, proponemos una serie de ejemplos resueltos donde se incluyen tablas de la base de datos descrita en capítulos anteriores. Algunos de ellos involucran incluso más de dos tablas. Se deja al lector la tarea de ir desarrollando las etapas antes descritas para cada uno de los ejemplos.

Ejemplo 6.17. *Encontrar la lista de los profesores (NRP y nombre) que imparten la asignatura BD1.*

Resultado:

$$\pi_{NRP,nom\_pro}(\sigma_{profesores.NRP=grupos.NRP} (profesores \times \sigma_{cod\_asig=''BD1'}(grupos)))$$

Ejemplo 6.18. *Encontrar los códigos de las asignaturas de las que está matriculado el alumno de nombre 'Luis Martinez Perez'.*

Resultado:

$$\pi_{cod\_asig}(\sigma_{alumnos.DNI=matricula.DNI} (matriculas \times \sigma_{nom\_alum='Luis\ Martinez\ Perez'}(alumnos)))$$

Ejemplo 6.19. *Encontrar los nombres de los profesores con categoría CU o TU que pertenecen al departamento de nombre Electrónica.*

Resultado:

$$\pi_{nom\_prof}(\sigma_{profesores.cod\_dep=departamentos.cod\_dep}$$
$$(\sigma_{categoria='CU' \vee categoria='TU'}(profesores)\times$$
$$\times\sigma_{nom\_dep='Electronica'}(departamentos)))$$

Ejemplo 6.20. *Encontrar los nombres de las asignaturas de las que está matriculado el alumno 'Luis Martinez Perez'.*

Resultado:

$$\pi_{nom\_asig}(\sigma_{alumnos.DNI=matricula.DNI \wedge matricula.cod\_asig=asignaturas.cod\_asig}$$
$$(matriculas \times \sigma_{nom\_alum='Luis\ Martinez\ Perez'}(alumnos) \times asignaturas))$$

Ejemplo 6.21. *Encontrar los nombres de los profesores que imparten prácticas en la asignatura 'Bases de Datos'. Entendemos que los grupos de prácticas son los grupos de tipo 'P'.*

Resultado:

$$\pi_{nom\_prof}(\sigma_{profesores.NRP=grupos.NRP \wedge grupos.cod\_asig=asignaturas.cod\_asig}$$
$$(\sigma_{tipo='P'}(grupos) \times \sigma_{nom\_asig='Bases\ de\ Datos'}(asignaturas) \times profesores))$$

Ejemplo 6.22. *Encontrar el nombre y el DNI de aquellos alumnos cuya provincia es 'Almeria' y que están matriculados de alguna asignatura de primer curso.*

Resultado:

$$\pi_{DNI,nom\_alum}(\sigma_{alumnos.DNI=matricula.DNI \wedge matricula.cod\_asig=asignaturas.cod\_asig}$$
$$(matriculas \times \sigma_{provincia='Almeria'}(alumnos) \times \sigma_{curso=1}(asignaturas)))$$

6.3.3. Conexión de una tabla consigo misma: Definición de alias

Existen situaciones en las que es necesario establecer conexiones entre los atributos de una misma tabla. Esta consultas surgen, por ejemplo, cuando se quieren obtener aquellas tuplas con el mismo valor en algún atributo que otra tupla concreta de la misma tabla.

Ejemplo 6.23. *Obtener el DNI y el nombre de aquellos alumnos que han nacido en la misma provincia que 'Luis Martinez Perez'.*

Para realizar esta consulta tendríamos que:

1. Seleccionar nom_alum='Luis Martinez Perez' sobre la tabla ALUMNOS:

$$\sigma_{alumnos.nomb\_alum='Luis\ Martinez\ Perez'}(alumnos)$$

2. Hacer el producto cartesiano de alumno con esta selección para poder igualar las provincias:

$$\sigma_{alumnos.nom\_alum='Luis\ Martinez\ Perez'}(alumnos) \times (alumnos)$$

Pero en este punto tenemos un problema, ya que no podemos distinguir si queremos referirnos a atributos de la tabla resultado de la selección o de la tabla alumnos completa, que participan en este producto cartesiano, ni siquiera cualificándolos con el nombre de la tabla, pues ambas proceden de la misma relación. Por ello se hace necesario poder hacer referencia a una tabla con un nombre distinto; es decir, *establecer un alias*. Para lograrlo, definimos un nuevo operador de la siguiente manera:

Definición 6.5. *Sea $R[A_1, ...A_n]$ una relación cualquiera y r una instancia de R, el operador* **redefinición aplicado a r**, *que notaremos por $\rho(r)$ nos permite asignar un nuevo nombre a r. De forma que*

$$\rho(r) = s$$

nos permite referirnos a r como s. Se dice entonces que s es un **alias** *de r.*

Con este nuevo operador podemos resolver el ejemplo 6.23 de la siguiente forma:

1. Definir un alias $\rho(alumnos) = alu$

2. Seleccionar nom_alum='Luis Martinez Perez' sobre la tabla ALUMNOS:

$$\sigma_{alumnos.nom\_alum='Luis\ Martinez\ Perez'}(alumnnos)$$

3. Hacer el producto cartesiano de ALU con esta selección e igualar las provincias en ALUMNOS y ALU para después proyectar sobre DNI y nom_alum. Obtenemos:

$$\pi_{alu.DNI,alu.nom\_alum}(\sigma_{alumnos.provincia=alu.provincia}$$

$$(\sigma_{alumnos.nom\_alum='Luis\ Martinez\ Perez'}(alumnos) \times (alu)))$$

El operador redefinición no se considera dentro del AR, ya que realmente no transforma una tabla. Es un mero recurso sintáctico para poder conectar una tabla consigo misma.

Ejemplos de la utilidad de estas conexiones son los siguientes:

Ejemplo 6.24. *Encontrar los nombres de los profesores que pertenecen a la misma área de conocimiento que 'Maria Lopez Ruiz'.*

Resultado:

$$\rho(profesores) = profes$$

$$\pi_{profes.nom\_prof}(\sigma_{profesores.area=profes.area}$$

$$(\sigma_{profesores.nom\_prof='Maria\ Lopez\ Ruiz'}(profesores) \times (profes)))$$

Ejemplo 6.25. *Encontrar el DNI y el nombre de aquellos alumnos de edad mayor o igual que la del alumno 'Luis Martinez Perez'.*

Resultado:

$$\rho(alumnos) = alu$$

$$\pi_{alu.DNI,alu.nom\_alum}(\sigma_{alumnos.edad \leq alu.edad}$$

$$(\sigma_{alumnos.nom\_alum='Luis\ Martinez\ Perez'}(alumnos) \times (alu)))$$

Ejemplo 6.26. *Encontrar aquellas asignaturas optativas que están en cursos superiores a la asignatura de nombre 'Bases de Datos'.*

Resultado:

$$\rho(asignaturas) = asis$$

$$\pi_{asis.nom\_asig}(\sigma_{asignaturas.curso<asis.curso}$$
$$(\sigma_{asignaturas.nom\_asig='Bases\ de\ Datos'}(asignaturas) \times \sigma_{asis.caracter='op'}(asis)))$$

Hay que hacer constar que el uso de alias no se limita al caso del producto cartesiano de una tabla consigo misma. En general, los alias serán necesarios cuando haya que distinguir atributos de una tabla que aparezca varias veces en expresiones complejas de AR.

6.3.4. La reunión natural

En una gran mayoría de los ejemplos que hemos desarrollado en torno a la Θ-reunión se observa que la propiedad de comparación es la igualdad y que esta se establece para "pegar" dos tablas a través de la conexión clave externa-clave primaria.

En este tipo de operaciones aparece siempre una columna repetida como mínimo, por lo que se hace necesaria una proyección que elimine las columnas repetidas. Esta forma de reunión es tan común que ha dado origen a una operación "especial", más concretamente, a una combinación de operaciones que tiene un símbolo específico para ella. Nos referimos a la **reunión natural** que pasamos a definir:

Definición 6.6. *Sean $R[A_1..A_n]$ y $S[B_1..B_m]$ dos relaciones tales que existen $\{A_i...A_j\} \subseteq \{A_1..A_n\}$ y $\{B_i...B_j\} \subseteq \{B_1..B_m\}$ de forma que $\forall k \in \{i..j\}$, $A_k = B_k$. Sean r y s instancias de R y S respectivamente. Definimos la reunión natural de r y s como:*

$$r \bowtie s = \pi_{(\{A_1..A_n\} \bigcup \{B_1..B_m\})-\{A_i...A_j\}}(\sigma_{(r.A_i=s.A_i)\wedge...\wedge(r.A_j=s.A_j)}(r \times s))$$

Es decir, el resultado de hacer la reunión bajo igualdad de los atributos que tienen el mismo nombre y eliminar, mediante proyección las columnas redundantes del resultado.

Ejemplo 6.27. *Supongamos las relaciones $R[A, B, C]$ y $S[B, C, D, E]$ siendo r y s instancias de estas tablas tal como aparece en la tabla 6.14, el resultado w de hacer $w = r \bowtie s$ aparece en la tabla 6.15.*

Ejemplos de utilización de esta operación, dentro de nuestra base de datos de trabajo, son los siguientes:

r=

A	B	C
a_1	b_1	c_1
a_2	b_2	c_2
a_3	b_1	c_1
a_4	b_1	c_1
a_4	b_2	c_2

s=

B	C	D	E
b_1	c_1	d_1	e_1
b_2	c_2	d_2	e_2
b_1	c_1	d_1	e_3
b_1	c_3	d_3	e_1
b_1	c_2	d_2	e_1

Tabla 6.14: Ejemplo de reunión natural: tablas que intervienen.

$r \bowtie s =$

A	B	C	D	E
a_1	b_1	c_1	d_1	e_1
a_1	b_1	c_1	d_1	e_3
a_3	b_1	c_1	d_1	e_1
a_3	b_1	c_1	d_1	e_3
a_2	b_2	c_2	d_2	e_2
a_4	b_2	c_2	d_2	e_2
a_4	b_1	c_1	d_1	e_1
a_4	b_1	c_1	d_1	e_3

Tabla 6.15: Ejemplo de reunión natural: resultados.

Ejemplo 6.28. *Obtener DNI y nombre de aquellos alumnos matriculados de la asignatura de código BDI que son becarios.*

Resultado:

$$\pi_{alumnos.DNI,nom\_alum}(\sigma_{beca='SI'}(alumnos) \bowtie \sigma_{cod\_asig='BDI'}(matriculas))$$

Ejemplo 6.29. *Encontrar la lista de los profesores (NRP y nombre) que imparten la asignatura BD1.*

Resultado:

$$\pi_{NRP,nom\_prof}(profesores \bowtie \sigma_{cod\_asig='BD1'}(grupos))$$

Ejemplo 6.30. *Encontrar los nombres de los profesores que imparten prácticas en la asignatura 'Bases de Datos'. Entenderemos que los grupos de ṕracticas son los grupos de tipo 'P'.*

Resultado:

$$\pi_{nom\_prof}(\sigma_{tipo='P'}(grupos) \bowtie$$

$$\bowtie \sigma_{nom\_asig='Bases\ de\ Datos'}(asignaturas) \bowtie profesores)$$

Como puede verse, los ejemplos planteados corresponden a soluciones alternativas a las preguntas que aparecen en los ejemplos 6.16, 6.17 y 6.21, donde se ha sustituido la correspondiente Θ-reunión por la reunión natural. Se recomienda al lector realizar la misma sustitución en aquellos ejemplos de Θ-reunión donde esto sea posible.

La reunión natural supone que los atributos comunes a las dos tablas tienen el mismo nombre e iguala "todos" los atributos con el mismo nombre en las tablas que intervienen en la operación.

Es importante tener en cuenta que, en el caso de que dos tablas tengan más de un atributo común y sólo se desee igualar uno de ellos, habrá que usar la Θ-reunión especificando el atributo que se desee.

6.4. Unión, intersección y diferencia

Pasamos ahora a definir el resto de los operadores que se basan en la idea de que una relación es un conjunto.

Definición 6.7. *Sean* $R[A_1..A_n]$, *y* $S[B_1..B_n]$ *dos relaciones tales que* $\{A_1..A_n\} \equiv \{B_1...B_n\}$ *y sean r y s instancias de R y S. Definimos:*

$$r \bigcap (\bigcup)(-) s = w$$

donde w es el resultado de hacer la intersección (unión) (diferencia) de r y s consideradas como conjunto de tuplas

Ejemplo 6.31. *Supongamos las relaciones* $R[A, B, C]$ *y* $S[A, B, C]$*, siendo* r *y* s *instancias de estas tablas, tal como aparecen en la figura 6.15. El resultado* w *de hacer* $w = r \bigcup (\bigcap)(-) s$ *se muestra en la figura 6.4.*