

Bases de Datos

Clase 2: Modelo Relacional - Álgebra Relacional

Hasta ahora

- Todo el mundo necesita manejar datos
- Salvo que queramos programar cosas que no tienen que ver con nuestro problema, conviene utilizar un DBMS
- Arquitectura de capas:
 - Usuarios finales ven modelo lógico
 - Sistema ejecuta las acciones
 - Usuario no necesita saber cómo el sistema ejecuta las acciones

DBMS

DBMS relacionales comerciales

- IBM DB2
- Microsoft SQL Server
- Oracle

DBMS

DBMS relacionales Open Source

- PostgreSQL - Es la que usaremos nosotros
- MySQL - Usada ampliamente en ambientes de producción
- SQLite - Base de datos pequeña, usada generalmente en contextos de apps móviles

DBMS

Otros DBMS

- Neo4J (Grafos)
- MongoDB (Documentos)
- Cassandra (Key Value - Column Store)
- Apache Jena (RDF)
- Redis (In memory Store)
- Base (Column Store)
- Titan DB (Grafos)
- ...

Modelos de datos

Modelo de Datos

- Un Modelo es una notación para escribir datos
- En este curso se verán en detalles dos modelos:
 - Relacional
 - Semiestructurados (key-value, XML, grafos)
- Modelo relacional es el modelo más usado en ambiente de producción, pero ha existido la necesidad de utilizar cada vez más modelos semiestructurados.

Modelo Relacional

Busca almacenar datos en tablas:

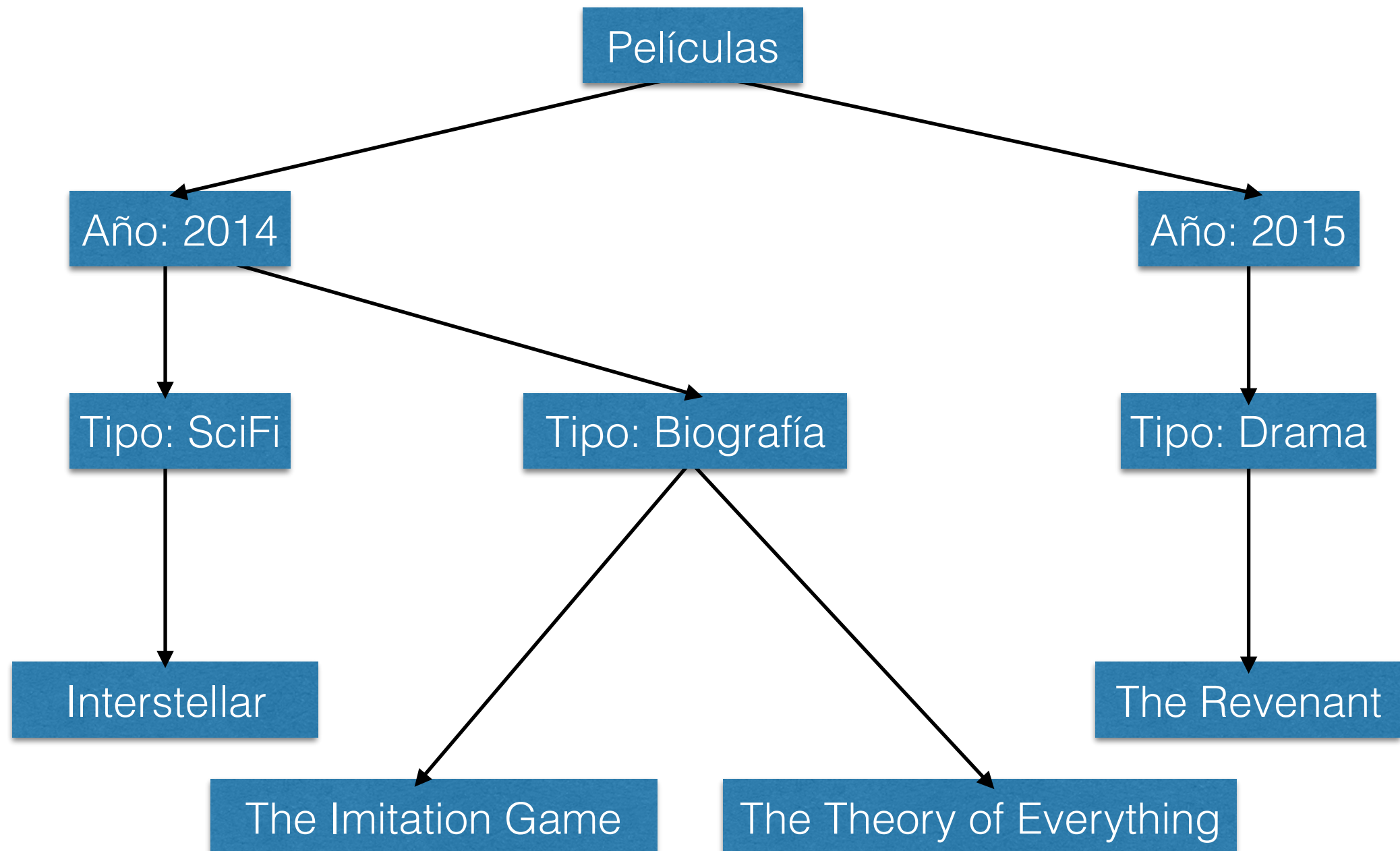
ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

Modelo Relacional

- Aunque parecen arreglos o listas (de Python por ejemplo) existen muchas diferencias
- Generalmente, se asume que están en Disco y no en Memoria
- No podemos hacer todo lo que queramos con esto
- Vamos a ver como manejar estos datos:
 - SQL
 - Álgebra Relacional

Datos Semiestructurados

Estructura Jerárquica



Datos Semiestructurados

XML

```
<Películas>
  <Año valor="2014">
    <Tipo valor="Biografía">
      <Película nombre="The Imitation Game" calificación="8.1">
      </Película>
      <Película nombre="The Theory of Everything" calificación="7.7">
      </Película>
    </Tipo>
    <Tipo valor="SciFi">
      <Película nombre="Interstellar" calificación="8.6">
      </Película>
    </Tipo>
  </Año>
  <Año valor="2015">
    <Tipo valor="Drama">
      <Película nombre="The Revenant" calificación="8.1">
      </Película>
    </Tipo>
  </Año>
</Películas>
```

Datos Semiestructurados

Key - Value

```
{
  "2014": {
    "Biografía": [
      { "nombre": "The Imitation Game", "calificación": 8.1 },
      { "nombre": "The Theory of Everything", "calificación": 7.7 }
    ],
    "SciFi": [
      { "nombre": "Interstellar", "calificación": 8.6 }
    ]
  },
  "2015": {
    "Drama": [
      { "nombre": "The Revenant", "calificación": 8.1 }
    ]
  }
}
```

Comparación

Ambos:

- Proveen solución para almacenar datos
- Son versátiles para modelar
- Ambos tienen lenguaje de consultas

Pero:

- Modelo relacional está definido por un **esquema**
- XML es más flexible, no está separado por un esquema

El modelo relacional al ser menos flexible es más simple pero también limitado

Otros Modelos

- Bases de Datos orientados a objetos
- Modelo Objeto - Relacional
- Bases de Datos de Grafos

Tenemos que aprender las diferencias y cuando usar qué!

Modelos Relacional

Modelo Relacional

Los datos se almacenan como tablas:

Películas

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

Distinguimos:

- Relaciones: a cada tabla le llamamos relación
- Atributos: son las columnas de la relación
- Tuplas: son las filas de la relación

Modelo Relacional

Películas

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

- En este caso tenemos la relación **Películas**
- Los atributos de la relación **Películas** son **ID_Película**, **Nombre_Película**, **Año**, **Categoría** y **Calificación**.

Modelo Relacional

Esquema

Para denominar relaciones escribimos su nombre y luego sus atributos entre paréntesis:

**Películas(id, nombre, año, categoría,
calificación)**

Modelo Relacional

Esquema

Un esquema es un conjunto de relaciones con sus atributos:

`Películas(id, nombre, año, categoría, calificación)`

`Actor(id, nombre, edad)`

`Actuó_en(id_actor, id_película)`

Modelo Relacional

Dominio

En la práctica, asumimos que cada atributo tiene un dominio (float, integer, string, date, ...)

```
Películas(id:int, nombre:string,  
año:int, categoría:string,  
calificación:float)
```

Modelo Relacional

Instancia

Una **instancia** de un **esquema** es un conjunto de tuplas para cada relación del esquema

Esto es un esquema:

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
-------------	-----------------	-----	-----------	---------------------

Modelo Relacional

Instancia

Una **instancia** de un **esquema** es un conjunto de tuplas para cada relación del esquema

Esto es una instancia:

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

Modelo Relacional

Restricciones de integridad

Son restricciones que imponemos a un esquema que todas las instancias deben satisfacer

La restricción más importante son las **llaves**

Un conjunto de atributos forma una **llave** en una **relación** si no permitimos que existan dos tuplas para esa relación con los mismos valores en todos los atributos de la llave, y no hay un subconjunto de esos atributos que cumpla esa condición.

Modelo Relacional

Ejemplo: Llaves

¿Cuál es la llave?

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	Batman	2005	Acción	8.3
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7
5	Batman	1995	Acción	5.4

Nota:

- Batman Begins (2005)
- Batman Forever (1995)

Modelo Relacional

Llaves

Cuando escribimos las relaciones subrayamos las llaves

`Películas(id, nombre, año, categoría,
calificación)`

Cómo consultar bases de datos

Ejemplos de Consultas

actores

id	nombre	edad
1	Leonardo DiCaprio	41
2	Matthew McConaughey	46
3	Daniel Radcliffe	27
4	Jessica Chastain	39
...

actuo_en

id_actor	id_pelicula
1	2
2	1
4	1
3	3
1	5
...	...

películas

id	nombre	año	categoría	calificación	director
1	Interstellar	2014	SciFi	8.6	C. Nolan
2	The Revenant	2015	Drama	8.1	A. Iñárritu
3	Harry Potter	2011	Fantasía	8.1	D. Yates
4	The Theory of Everything	2014	Biografía	7.7	J. Marsh
5	Inception	2010	Adventure	8.8	C. Nolan
...

Ejemplos de Consultas

Ejemplo 1

Liste el nombre de todos los actores

nombre
Leonardo DiCaprio
Matthew McConaughey
Daniel Radcliffe
Jessica Chastain

Ejemplos de Consultas

Ejemplo 2

Liste el nombre y la calificación de todas las películas

nombre	calificacion
Interstellar	8.6
The Revenant	8.1
Harry Potter	8.1
The Theory of Everything	7.7
Inception	8.8

Ejemplos de Consultas

Ejemplo 3

Liste el nombre y la calificación de todas las películas con calificación inferior a 8.5

nombre	calificacion
The Revenant	8.1
Harry Potter	8.1
The Theory of Everything	7.7

Ejemplos de Consultas

Ejemplo 4

Liste el todas las películas de Nolan

id	nombre	año	categoria	calificacion	director
1	Interstellar	2014	SciFi	8.6	C. Nolan
5	Inception	2010	Adventure	8.8	C. Nolan

Ejemplos de Consultas

Ejemplo 5

Liste todos los id de los actores de la película “Interstellar”

id
2
4

Ejemplos de Consultas

Ejemplo 6

Liste cada actor junto a todas las películas en las que ha actuado

id	nombre	nombre_pelicula
1	Leonardo DiCaprio	The Revenant
1	Leonardo DiCaprio	Inception
2	Matthew McConaughey	Interstellar
3	Daniel Radcliffe	Harry Potter
4	Jessica Chastain	Interstellar

Ejemplos de Consultas

Ejemplo 7

Liste todas las películas en que actúe Leonardo DiCaprio y que sean dirigidas por C. Nolan

nombre

Inception

Ejemplos de Consultas

Ejemplo 8

Liste todas las películas y la calificación en que actúe Leonardo DiCaprio o que sean dirigidas por C. Nolan

nombre	calificacion
Interstellar	8.6
The Revenant	8.1
Inception	8.8

Ejemplos de Consultas

Ejemplo 9

Liste el nombre de todos los actores y directores

nombre
Leonardo DiCaprio
Matthew McConaughey
Daniel Radcliffe
Jessica Chastain
C. Nolan
A. Iñárritu
D. Yates
J. Marsh

Ejemplos de Consultas

Ejemplo 10

Liste el nombre de todos los actores dirigidos por C. Nolan y A. Iñárritu

id	nombre	edad
1	Leonardo DiCaprio	41

Ejemplos de Consultas

¿Qué podemos concluir?

- Los resultados de las consultas también son tablas
- Parecen haber operaciones en común

Álgebra relacional de selección, proyección y unión

- Lenguaje teórico
- Posee un conjunto de operadores que como input toman tablas, y como output devuelven tablas

$$\pi, \sigma, \cup, \times$$

- Durante el curso veremos cómo este lenguaje forma los cimientos de todos los lenguajes de consulta

Operadores

Proyección

Sea R una relación, entonces $\pi_{a_1, \dots, a_n}(R)$ es una nueva relación que deja sólo a los atributos a_1, \dots, a_n de R

Liste todos los actores:

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

Liste el nombre y la calificación de todas las películas:

$$\pi_{\text{nombre}, \text{calificacion}}(\text{peliculas})$$

Selección

Sea R una relación, entonces $\sigma_{condición}(R)$ es una nueva relación que deja sólo a las tuplas (filas) que satisfacen la condición

Las condiciones pueden ser:

$<, \leq, \geq, >, =, \neq$

Y se pueden combinar con:

\wedge, \vee

Selección

Liste todas las películas de C. Nolan:

$$\sigma_{director="C.Nolan"}(películas)$$

Liste todas las películas con calificación inferior a 8.5 y dirigidas por J. Marsh

$$\sigma_{calificacion<8.5 \wedge director="J.Marsh"}(películas)$$

Liste el nombre y la calificación de todas las películas con calificación inferior a 8.5

$$\pi_{nombre,calificacion}(\sigma_{calificacion<8.5}(películas))$$

Unión

Sean R_1, R_2 relaciones con la misma cantidad de atributos y del mismo tipo, entonces $R_1 \cup R_2$ es una nueva relación que contiene la unión de las tuplas de R_1 y R_2

Liste el nombre de todos los actores y directores

$$\pi_{nombre}(actores) \cup \pi_{director}(peliculas)$$

Observación: Las relaciones son conjuntos de tuplas, por lo que si una tupla está en R_1 y R_2 , la unión contiene a la tupla una sola vez

Renombrando atributos

Para cambiar nombres de atributos en una relación usamos el operador ρ

Para cambiar:

`películas(id, nombre, año, categoria, calificacion, director)` por
`películas(id, name, year, category, rating, director)`

$$\rho((\textit{nombre} \rightarrow \textit{name}, \textit{año} \rightarrow \textit{year}, \\ \textit{categoria} \rightarrow \textit{category}, \textit{calificacion} \rightarrow \textit{rating}), \textit{películas})$$

Renombrando relaciones

Para cambiar nombres de relaciones (y usarlas más tarde) usamos el operador ρ

Guardamos en `actores_jovenes` todos los actores de menos de 30 años:

$$\rho(\textit{actores_jovenes}, \sigma_{\textit{edad} < 30}(\textit{actores}))$$

Luego si queremos consultar los nombres de esos actores:

$$\pi_{\textit{nombre}}(\textit{actores_jovenes})$$

Producto Cruz

Nos falta cruzar información entre tablas

El operador \times permite hacer el producto cartesiano de dos relaciones

R_1			$R_1 \times R_2$			
A	B		R_2	A	C	D
a_1	b_1	\times	a_1	c_1	d_1	$=$
a_2	b_2		a_2	c_2	d_2	

$R_1.A$	$R_1.B$	$R_2.A$	$R_2.C$	$R_2.D$
a_1	b_1	a_1	c_1	d_1
a_1	b_1	a_2	c_2	d_2
a_2	b_2	a_1	c_1	d_1
a_2	b_2	a_2	c_2	d_2

Observación: La cardinalidad está dada por $|R_1 \times R_2| = |R_1| \cdot |R_2|$

Producto Cruz

Liste todos los id de los actores de la película “Interstellar”

1) Hacemos el producto cruz de `peliculas` y `actuo_en`

Producto Cruz

películas × actuo_en

pelicula. id	pelicula.nom bre	pelicula. año	pelicula.c ategoria	pelicula.cali ficacion	pelicula. director	actuo_e n.id_act or	actuo_en.id _pelicula
1	Interstellar	2014	SciFi	8.6	C. Nolan	1	2
1	Interstellar	2014	SciFi	8.6	C. Nolan	2	1
1	Interstellar	2014	SciFi	8.6	C. Nolan	4	1
1	Interstellar	2014	SciFi	8.6	C. Nolan	3	3
1	Interstellar	2014	SciFi	8.6	C. Nolan	1	5
2	The Revenant	2015	Drama	8.1	A. Iñárritu	1	2
2	The Revenant	2015	Drama	8.1	A. Iñárritu	2	1
2	The Revenant	2015	Drama	8.1	A. Iñárritu	4	1
2	The Revenant	2015	Drama	8.1	A. Iñárritu	3	3
2	The Revenant	2015	Drama	8.1	A. Iñárritu	1	5

...

...

...

...

...

...

Producto Cruz

Liste todos los id de los actores de la película “Interstellar”

2) Filtramos cuando `pelicula.id` sea igual a `actuo_en.id_pelicula`

Producto Cruz

$$\sigma_{peliculas.id=actuo_en.id_pelicula}(peliculas \times actuo_en)$$

peliculas .id	peliculas.no mbre	pelicula s.año	peliculas. categoria	peliculas.ca lificacion	peliculas .director	actuo_e n.id_act or	actuo_en.id _pelicula
1	Interstellar	2014	SciFi	8.6	C. Nolan	2	1
1	Interstellar	2014	SciFi	8.6	C. Nolan	4	1
2	The Revenant	2015	Drama	8.1	A. Iñárritu	1	2
3	Harry Potter	2011	Fantasía	8.1	D. Yates	3	3
5	Inception	2010	Adventure	8.8	C. Nolan	1	5

Producto Cruz

Liste todos los id de los actores de la película “Interstellar”

3) Filtramos según el id de la película “Interstellar”:

Producto Cruz

$$\sigma_{\text{peliculas.id}=1} \left(\sigma_{\text{peliculas.id}=\text{actuo_en.id_pelicula}} \left(\text{peliculas} \times \text{actuo_en} \right) \right)$$

peliculas .id	peliculas.no mbre	pelicula s.año	peliculas. categoria	peliculas.ca lificacion	peliculas .director	actuo_e n.id_act or	actuo_en.id _pelicula
1	Interstellar	2014	SciFi	8.6	C. Nolan	2	1
1	Interstellar	2014	SciFi	8.6	C. Nolan	4	1

Join

Liste cada actor junto a todas las películas en las que ha actuado

Claramente debemos usar dos productos cruz y luego usar selección.

Vamos a utilizar el “operador” Join \bowtie . En realidad no es un operador, pues es definible con selección y producto cruz:

$$R_1 \bowtie_{condicion} R_2 = \sigma_{condicion}(R_1 \times R_2)$$

Join

Liste todos los id de los actores de la película “Interstellar”

$$\sigma_{\text{pelicula.id}=1}(\text{actuo_en} \bowtie_{\text{peliculas.id=actuo_en.id_pelicula}} \text{peliculas})$$

Join

Liste cada actor junto a todas las películas en las que ha actuado

$$\pi_{actores.nombre, peliculas.nombre}(\text{actuo_en} \bowtie_{peliculas.id=actuo_en.id_pelicula} \text{peliculas} \bowtie_{actores.id=actuo_en.id_actor} actores)$$

SQL

Natural Join

Cuando los atributos en ambas relaciones tienen el mismo nombre, es posible no indicar la condición:

Si las relaciones son: $R(a, b)$, $S(b, c)$

$$R \bowtie S$$

¿Qué pasa con esto?

Liste los actores dirigidos por C. Nolan y A. Iñárritu

$$\begin{aligned} & \rho(\text{peliculas_join}, \\ & \text{actuo_en} \bowtie_{\text{peliculas.id=actuo_en.id_pelicula}} \\ & \text{peliculas} \bowtie_{\text{actores.id=actuo_en.id_actor}} \\ & \text{actores}) \\ & \pi_{\text{actores.nombre}, \text{peliculas.nombre}} (\\ & \sigma_{\text{pelicula.director} = \text{"C.Nolan"} \wedge \text{pelicula.director} = \text{"A.Iñárritu"}} (\\ & \text{peliculas_join})) \end{aligned}$$

La consulta anterior no funciona, ¿por qué?

Intersección

Liste el nombre de todos los actores dirigidos por C. Nolan y A. Iñárritu

$$\rho(\text{peliculas_join}, \\ \text{actuo_en} \bowtie_{\text{peliculas.id=actuo_en.id_pelicula}} \\ \text{peliculas} \bowtie_{\text{actores.id=actuo_en.id_actor}} \\ \text{actores})$$

$$\rho(\text{peliculas_nolan}, \\ \pi_{\text{actores.id}, \text{actores.nombre}}(\\ \sigma_{\text{pelicula.director}=\text{"C.Nolan"}}(\\ \text{peliculas_join}))$$

$$\rho(\text{peliculas_inarritu}, \\ \pi_{\text{actores.id}, \text{actores.nombre}}(\\ \sigma_{\text{pelicula.director}=\text{"A.Iñárritu"}}(\\ \text{peliculas_join}))$$

$$\pi_{\text{nombre}}(\text{peliculas_nolan} \cap \text{peliculas_inarritu})$$

Intersección

Sean las relaciones R_1, R_2 , ambas con los mismos atributos, su intersección puede definirse con los operadores definidos anteriormente

$$\rho(R, R_1 \bowtie_{R_1.a_1=R_2.a_1 \wedge \dots \wedge R_1.a_n=R_2.a_n} R_2)$$

$$R_1 \cap R_2 = \pi_{R_1.a_1 \wedge \dots \wedge R_n.a_n}(R)$$

¿Qué pasa con esto?

Liste el nombre de todos los actores dirigidos por C. Nolan
y no por A. Iñárritu

No podemos realizar esta consulta!

Consultas monótonas

Sea E una expresión de álgebra relacional sobre un esquema S

E es monótona si para toda instancia I, J sobre S , si $I \subseteq J$, entonces se tiene $E(I) \subseteq E(J)$

Si aumenta el tamaño de mi instancia, entonces el resultado de mi consulta no disminuye!

Consultas monótonas

Teorema: Toda consulta usando los operadores

$$\rho, \times, \sigma, \pi, \cup$$

es monótona

Diferencia

Sean las relaciones R_1, R_2 , ambas con los mismos atributos, su diferencia $R_1 - R_2$ es una nueva relación que contiene la diferencia (de conjuntos) entre las tuplas de ambas relaciones

Diferencia

Liste el nombre de todos los actores dirigidos por C. Nolan y no por A. Iñárritu

$$\rho(\text{peliculas_join}, \\ \text{actuo_en} \bowtie_{\text{peliculas.id=actuo_en.id_pelicula}} \\ \text{peliculas} \bowtie_{\text{actores.id=actuo_en.id_actor}} \\ \text{actores})$$
$$\rho(\text{peliculas_nolan}, \\ \pi_{\text{actores.id}, \text{actores.nombre}}(\\ \sigma_{\text{pelicula.director}=\text{"C.Nolan"}}(\\ \text{peliculas_join}))$$
$$\rho(\text{peliculas_inarritu}, \\ \pi_{\text{actores.id}, \text{actores.nombre}}(\\ \sigma_{\text{pelicula.director}=\text{"A.Iñárritu"}}(\\ \text{peliculas_join}))$$
$$\pi_{\text{nombre}}(\text{peliculas_nolan} - \text{peliculas_inarritu})$$