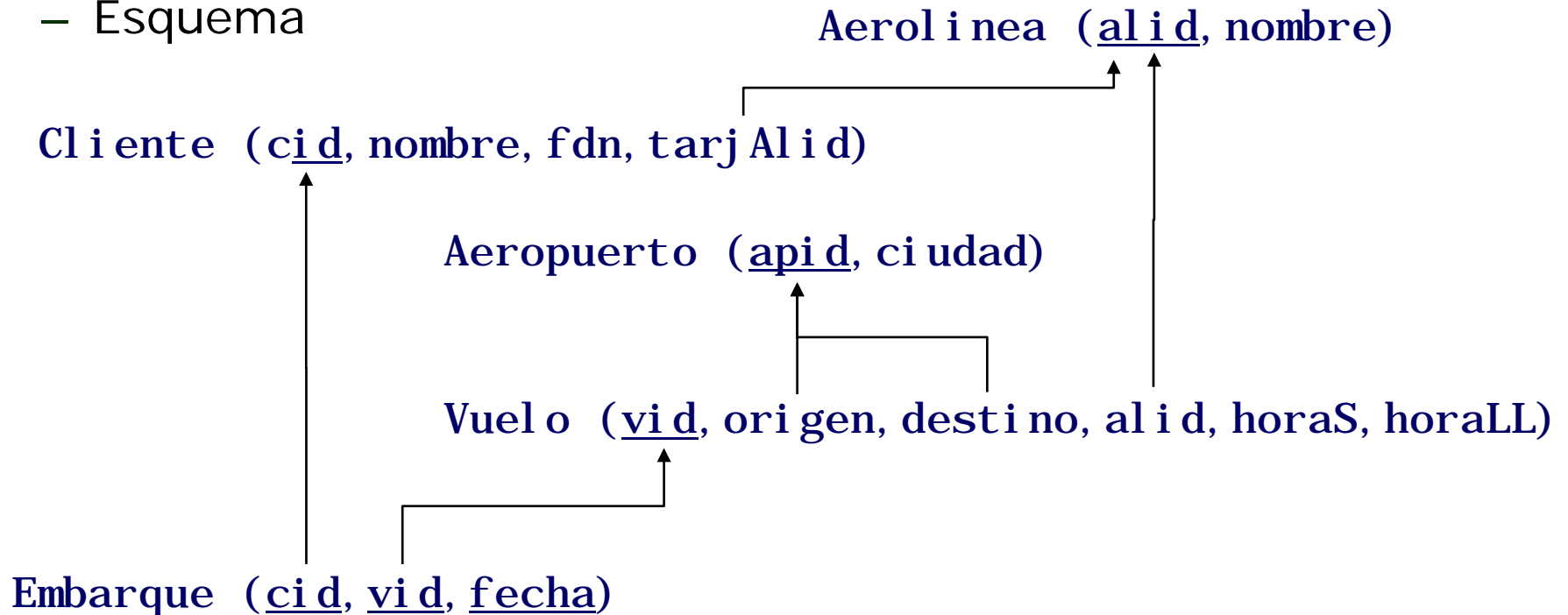


Ejercicio SQL: Viajes en avión

– Esquema



...

- Todos los vuelos realizados por 'García'

```
select C.nombre, E.vi d  
from Cliente C, Embarque E  
where C.cid=E.cid and C.nombre like 'García%';
```

...

- Número de vuelos de la compañía 'Qantas'

```
select AL.nombre, count(*)  
from Aerolinea AL, Vuelo V  
where AL.alid=V.alid and AL.nombre like 'Qantas%';
```

...

- Compañía (alid) con el máximo número de vuelos

```
select V.alid, count(*) as nvu
from Vuelo V
group by V.alid
having count(*) >= ALL
    (select count(*) as nvu
     from Vuelo V
     group by V.alid);
```

```
with NVuelos as (select V.alid, count(*) as nvu
                  from Vuelo V
                  group by V.alid)
select NV.alid, NV.nvu
from NVuelos NV
where NV.nvu >= ALL (select NV2.nvu from NVuelos NV2);
```

WITH es SQL-99

...

- 5 compañías con el mayor número de vuelos

```
with NVuelos as (select V.alid, count(*) as nvu
                  from Vuelo V
                  group by V.alid)
select NV.alid, NV.nvu
from NVuelos NV
where 5 > (select count(*)
           from Nvuelos NV2
           where NV2.nvu > NV.nvu);
```

...

- Pares origen-destino (apids) con al menos 2 vuelos, ordenado por nº vuelos

```
select V.origen, V.destino, count(*) as nvu
from Vuelo V
group by V.origen, V.destino
having count(*) >= 2
order by nvu desc
```

...

- Encontrar todos los vuelos (vuelo-día) vacíos

```
select V.vid, F.fecha
from Vuelo V,
      (select distinct E.fecha from Embarque E) F
where (V.vid, F.fecha) NOT IN (select E.vid, E.fecha
                               from Embarque E)
```

...

- Personas (cid,nombre) que han viajado en su cumpleaños
Pista: utilizar `extract(<day|month|year> from c.fdn)`

```
select distinct C.cid, C.nombre
from Cliente C, Embarque E
where E.cid=C.cid and
      extract(day from E.fecha) = extract(day from C.fdn) and
      extract(month from E.fecha) = extract(month from C.fdn)
```


...

- Compañía con el máximo número de viajeros

```
with Nviajeros as (  
    select V.alid, count(distinct E.cid) as nvi  
    from Vuelo V, Embarque E  
    where V.vid=E.vid  
    group by V.alid )  
select NV.alid, NV.nvi  
from Nviajeros NV  
where NV.nvi >= ALL (select NV2.nvi from Nviajeros NV2);
```

...

- Personas (cid,nombre) que no tienen tarjeta de Qantas, pero es en la que más han viajado

```
with NVuelos as (  
    select E.cid, V.alid, count(*) as nvu  
    from Vuelo V NATURAL JOIN Embarque E  
    group by E.cid, V.alid )  
  
select C.cid, C.nombre  
from Cliente C, NVuelos NV  
where C.cid=NV.cid and  
    NV.nvu >= ALL (select NV2.nvu  
                    from NVuelos NV2  
                    where NV2.cid = NV.cid) and  
    NV.alid = 'QFA' and  
    C.tarjAlid <> 'QFA';
```

...

- Personas que visitaron más ciudades en un periodo (fini,ffin)

```
with APVisitados as (  
    select distinct E.cid, AP.ciudad  
    from Embarque E, Vuelo V, Aeropuerto AP  
    where E.vid=V.vid and E.fecha>=fini and  
           E.fecha<=ffin and  
           (V.origen=AP.apid OR  
            V.destino=AP.apid) )
```

```
With NCiudades as (  
    select APV.cid, count(*) as nciu  
    from APVisitados APV  
    group by APV.cid )
```

```
select NC.cid, NC.nciu  
from NCiudades NC  
where NC.nciu >= ALL (select NC2.nciu from NCiudades NC2);
```

...

- Lista ordenada de aeropuertos y su rango; rango=nº total de vuelos con origen en el aeropuerto; si dos empatan deben tener el mismo rango, y el siguiente número no aparecer

```
with NVuelos as (  
    select V.origen, count(*) as nvu  
    from Vuelos V  
    group by V.origen )  
  
select NV.apid, NV.nvu, (select count(*)+1  
                        from NVuelos NV2  
                        where NV2.nvu > NV.nvu) as rank  
from NVuelos NV  
order by rank desc
```