Carlos Alvarado
Marc Badia

Query 1, 3, 4 (key = orderkey)

orderkey

c_mktsegment

o_order_date

o_ship_priority

n_name

r_name

lineitems:

- l_extendedprice
- l_discount
- l_returnflag
- l_linestatus
- l_quantity
- l_tax

Utilizamos esta colección principalmente porque en las consultas 1, 3 y 4 se utilizan repetidamente las clases Order, Customer y Lineitem. Utilizamos orderkey como clave ya que la clase order contiene un solo customer y muchos (1..*) lineitems, por lo que el customer queda implícito dado el orderkey y ponemos sus atributos al mismo nivel que los de order (c_mktsegment). En cambio los lineitems los ponemos como subdocumento, por lo que dado un orderkey, tenemos un array de lineitems. Podríamos haberlo colocado lineitems al mismo nivel que order y customer con una key doble (orderkey, custkey), pero nos hemos decantado por la otra opción debido a que creemos que es la más eficiente. Tenemos SF*6.000.000 en la clase lineitems i SF*1.500.00 en orders por lo que tendriamos SF*6.000.000 ficheros si lo hicieramos con lineitems en clave primaria, en cambio solo tenemos SF*1.500.000 ficheros con nuestra manera y una media de 4 lineitems por order.

En la consulta 3 utilizamos las clases de lineitems, orders y lineitems. Por todo lo que hemos explicado anteriormente la mejor opción es hacerlo en una colección donde orderkey es la clave.

En la consulta 4, es lo mismo pero añadiendo nation y region, por lo que añadimos los atributos necesarios a la colección en el nivel de order (una order tiene un customer, que tiene una nation que tiene una region).

Por último, en la consulta 1 solo utilizamos lineitems. Esta consulta podría hacerse con los lineitems, orders y clients al mismo nivel. SIn embargo, como con las otras dos es mejor con order como nivel superior, decidimos mantener ese formato, para que la eficiencia media sea òptima.

Carlos Alvarado
Marc Badia

Query 2 v2 (key = suppkey)

suppkey
s_acctbal
s_name
s_address
s_phone
s_comment
n_name
r_name
parts:
    partkey
    p_mfgr
    p_size
    p_type
    ps_supplycost

En este momento solo queda la query 2 libre, que utiliza las clases part, supplier, partsupp y nation y region en menor medida, por lo que nos fijaremos más en las tres primeras.
Al principio veíamos tres opciones viables:

- Todas al mismo nivel con (partkey, suppkey) como clave.
- Part como nivel superior con clave partkey y supplier y partsupp como subdocumento.
- Supplier como nivel superior con clave suppkey y part y partsupp como subdocumento.

Descartamos la primera opción porque tendríamos SF*800.000 ficheros, que son muchas más que con las otras opciones.
En la segunda opción tendríamos SF*200.000 ficheros en los cuales se repetirán los suppliers, por lo que es ineficiente des del punto de vista espacial y de procesamiento.
Por lo que nos quedamos con la tercera, con SF*10.000 ficheros en los cuales tendremos subdocumentos de 20 elementos de media, en los que la lógica nos dice que no se repetirán ya que no tendría sentido que un supplier tuviera partes repetidas (en cambio tiene sentido que una parte tenga varios suppliers).
Además la query accede más a los atributos de supplier que a los demás.