

Data Warehousing

Practical Task # 04 Materialized view & OLAP

Task 4c

1. Suppose there are 2 tables named Employee and Department. The Employee table contains 1 million records and department table contains 20 records. How to fetch the Employees associated with that department in less time, and how performance can be optimized using materialized view.
2. Explore materialized views that are based on aggregation of data. Suppose we build a materialized view on the base tables

Classes (class, type, country, numGuns, bore, displacement)

Ships (name, class, launched)

From our earlier battleships example, as follows:

```
CREATE MATERIALIZED VIEW ShipStats AS
```

```
SELECT country, AVG (displacement),
```

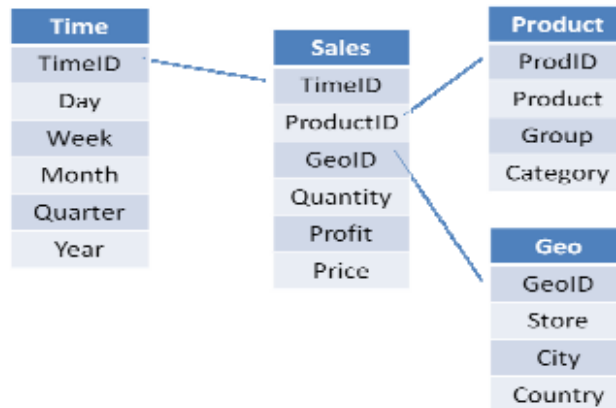
```
COUNT (*) FROM classes,
```

```
Ships WHERE Classes.class = Ships.class
```

```
GROUP BY country;
```

What modifications to the base tables Classes and Ships would require a modification of the materialized view? How would you implement those modifications incrementally?

3. Fill in with records the tables described in the schema from Fig1.
 - a. Write an SQL select query, which applies the Cube operator on all 3 dimensions, Cube (Week, Store, Group) of the database in Fig1 with the records you have filled above, and then present the result of the query as a table.
 - b. Write an SQL query which retrieves the top 3 selling product groups in February, in Braunschweig.



4. Consider a star schema with a fact table for sales, and 3 dimensions, the Geo, Time and Product dimension. The fact table has 5 integer (4 byte) attributes (1 primary key, 3 foreign keys, and the sales amount). The Geo table has 4 attributes: 1 primary key (of 4 bytes), and 3 string attributes (each of 20 bytes). The Product table has 4 attributes: 1 primary key (of 4 bytes), and 3 string attributes (each of 20 bytes). The Time table has 4 attributes: 1 primary key (of 4 bytes), one date (of 4 bytes) one string for the month (10 bytes) and the year (2 bytes).

Consider the case of a low selectivity percentage of about 50%, for a fact table of 10 million records, Geo dimension with 2000 records, Product dimension with 2000 records and Time dimension with 260 records. (Express all the intermediate results in MB, GB, or TB by case).

- How big the temporary results in the case of a dimensional cross product are join strategy? (Explain how you have reached these results).
- How big are the intermediate results in the case of the (IBM) semi-join strategy? (Explain how you have reached these results). Consider here a 32 bit based architecture with pointers to the data of 4 bytes.