Experiment No.2
Apply OLAP operations
Name: Sumit Metkari
Class: SE-2
Roll No. : 32
Date of Performance: 23-07-25
Date of Submission: 30-07-25

Aim: To implement Perform OLAP Operations

Objective: Develop a program to implement OLAP operations

Theory:

Online analytical processing, or **OLAP** is an approach to answering multi-dimensional analytical (MDA) queries swiftly in computing OLAP is part of the broader category of business intelligence, which also encompasses relational database, report writing and data mining. Typical applications of OLAP include business reporting for sales, marketing, management reporting, business process management (BPM), budgeting and forecasting, financial reporting and similar areas, with new applications coming up, such as agriculture. The term OLAP was created as a slight modification of the traditional database term online transaction processing (OLTP).

OLAP tools enable users to analyze multidimensional data interactively from multiple perspectives. OLAP consists of three basic analytical operations: consolidation (roll-up), drill-down, and slicing and dicing. Consolidation involves the aggregation of data that can be accumulated and computed in one or more dimensions. For example, all sales offices are rolled up to the sales department or sales division to anticipate sales trends. By contrast, the drill-down is a technique that allows users to navigate through the details. For instance, users can view the sales by individual products that make up a region's sales. Slicing and dicing is a feature whereby users can take out (slicing) a specific set of data of the OLAP cube and view (dicing) the slices from different viewpoints. These viewpoints are sometimes called dimensions (such as looking at the same sales by salesperson or by date or by customer or by product or by region, etc.)
The following are different types of OLAP models:

1. MOLAP (Multidimensional OLAP)

2. ROLAP (Relational OLAP)

3. HOLAP (Hybrid OLAP)

4. DOLAP (Desktop OLAP)

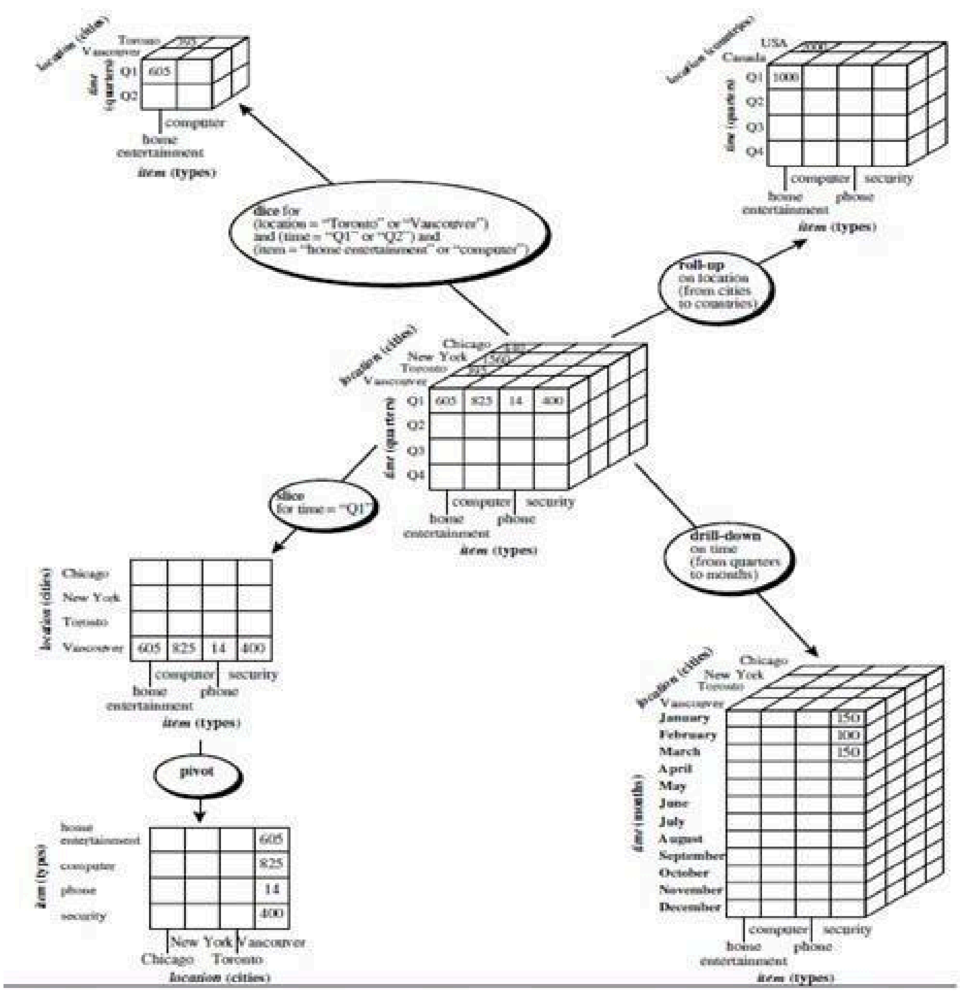

Figure 1: OLAP Operations

Roll-up: The roll-up operation (also called the drill-up operation by some vendors) performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension or by dimension reduction. Figure 1 shows the result of a roll-up operation performed on the central cube by climbing up the concept hierarchy for location given in Figure 1. This hierarchy was defined as the total order "street < city < province or state < country." The roll-up operation shown aggregates the data by ascending the location hierarchy from the level of city to the level of country. In other words, rather than grouping the data by city, the resulting cube groups the data by country. When roll-up is performed by dimension reduction, one or more dimensions are removed from the given cube. For example, consider a sales data cube containing only the two dimensions location and time. Roll-up may be performed by removing, say, the time dimension, resulting in an aggregation of the total sales by location, rather than by location and by time.

Drill-down: Drill-down is the reverse of roll-up. It navigates from less detailed data to more detailed data. Drill-down can be realized by either stepping down a concept hierarchy for a dimension or introducing additional dimensions. Figure 1 shows the result of a drill-down operation performed on the central cube by stepping down a concept hierarchy for time defined as "day < month < quarter < year." Drill-down occurs by descending the time hierarchy from the level of quarter to the more detailed level of month. The resulting data cube details the total sales per month rather than summarizing them by quarter. Because a drill-down adds more detail to the given data, it can also be performed by adding new dimensions to a cube. For example, a drill-down on the central cube of Figure 1 can occur by introducing an additional dimension, such ascustomer group.

Slice and dice: The slice operation performs a selection on one dimension of the given cube, resulting in a subcube. Figure 1 shows a slice operation where the sales data are selected from the central cube for the dimension time using the criterion time = "Q1". The dice operation defines a subcube by performing a selection on two or more dimensions. Figure 1 shows a dice operation on the central cube based on the following selection criteria that involve three dimensions: (location = "Toronto" or "Vancouver") and (time = "Q1" or "Q2") and (item = "home entertainment" or "computer").

Pivot (rotate): Pivot (also called rotate) is a visualization operation that rotates the data axes in view in order to provide an alternative presentation of the data. Figure 1 shows a pivot operation where the item and location axes in a 2-D slice are rotated.

Code and output:

```
-- 1. Total profit by movie, quarter, director
SELECT
   m.movie_title,
   t.quarter,
   d.director_name,
   SUM(f.profit) AS total_profit
FROM
   fact_movie_performance f
JOIN dim_movie m ON f.movie_key = m.movie_key
JOIN dim_director d ON f.director_key = d.director_key
JOIN dim_time t ON f.time_key = t.time_key
GROUP BY
   m.movie_title, t.quarter, d.director_name;

-- 2. Average rating for Q4 by genre and director
SELECT
   g.genre_name,
   d.director_name,
   t.quarter,
   AVG(f.rating) AS avg_rating
FROM
   fact_movie_performance f
JOIN dim_genre g ON f.genre_key = g.genre_key
JOIN dim_director d ON f.director_key = d.director_key
JOIN dim_time t ON f.time_key = t.time_key
WHERE
   t.quarter = 'Q4'
GROUP BY
   g.genre_name, d.director_name, t.quarter;

-- 3. Total box office revenue for Q3 and director "Christopher Nolan" by movie
SELECT
   m.movie_title,
   t.quarter,
   d.director_name,
   SUM(f.box_office_revenue) AS total_revenue
FROM
```

```
    fact_movie_performance f
JOIN dim_movie m ON f.movie_key = m.movie_key
JOIN dim_director d ON f.director_key = d.director_key
JOIN dim_time t ON f.time_key = t.time_key
WHERE
    t.quarter = 'Q3'
    AND d.director_name = 'Christopher Nolan'
GROUP BY
    m.movie_title, t.quarter, d.director_name;

-- 4. Yearly total box office revenue
SELECT
    t.year,
    SUM(f.box_office_revenue) AS yearly_revenue
FROM
    fact_movie_performance f
JOIN dim_time t ON f.time_key = t.time_key
GROUP BY
    t.year;

-- 5. Monthly total profit
SELECT
    t.month,
    SUM(f.profit) AS monthly_profit
FROM
    fact_movie_performance f
JOIN dim_time t ON f.time_key = t.time_key
GROUP BY
    t.month;
```

Output:

movie_title	quarter	director_name	total_profit
Inception	Q3	Christopher Nolan	669000000.00
KGF	Q4	Prashanth Neel	170000000.00

genre_name	director_name	quarter	avg_rating
Drama	Prashanth Neel	Q4	8.50000

movie_title	quarter	director_name	total_revenue
Inception	Q3	Christopher Nolan	829000000.00

year	yearly_revenue
2010	829000000.00
2018	250000000.00

month	monthly_profit
July	669000000.00
December	170000000.00

Conclusion: Comment on the usefulness of the OLAP queries on your application.

OLAP queries are highly useful in the application as they enable fast, multidimensional analysis of large datasets. They help in generating detailed reports, identifying trends, and making data-driven decisions efficiently, improving overall business intelligence and performance.