OLAP Operations

- OLAP operations in multidimensional data.
  - Roll – up
  - Drill – down
  - Slice
  - Dice
  - Pivoting (rotate)

OLAP Operations

- Roll – up : Aggregation to higher level
  - Perform aggregation on a data cube by.
  - Climbing up a concept hierarchy for a dimension
  - Example : In a given hierarchy for location dimension

    "street< city < province or state < country."
  - In order to find aggregate of sales in country
    - The roll-up operation aggregates the data by ascending the *location hierarchy from the level of city to the level of country.*

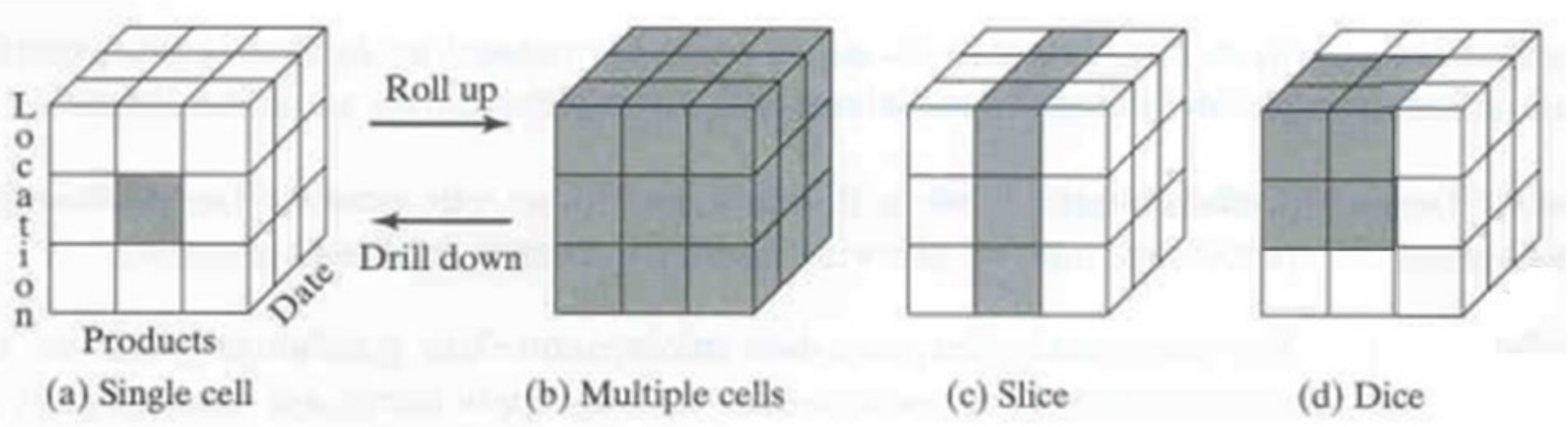
OLAP Operations

- Drill – down : Recalculation with more details
- Reverse of roll-up
- Navigates from less detailed data to more detailed data by
  - Stepping down a concept hierarchy for a dimension
  - Introducing additional dimensions

OLAP Operations

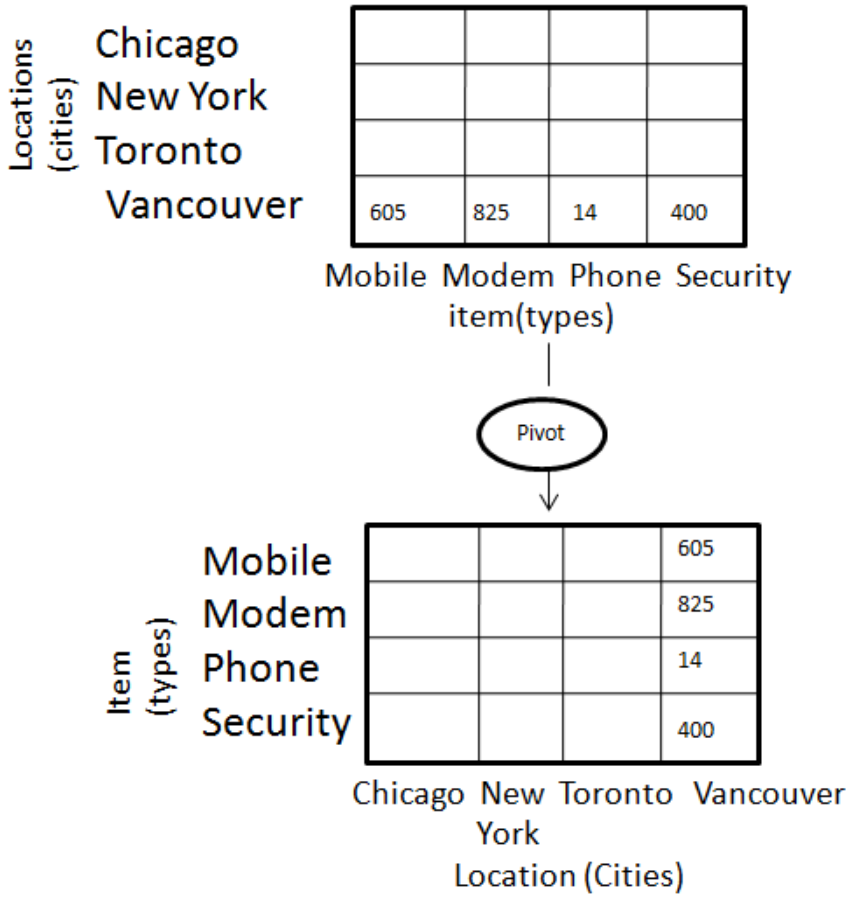
- Slice
  - The slice operation selects one particular dimension from a given cube and provides a new sub-cube
- Dice
  - Defines a sub-cube by performing a selection on two or more dimensions
- Pivoting
  - The pivot operation is also known as rotation.
  - It rotates the data axes in view in order to provide an alternative presentation of data.

OLAP Operations



OLAP Operations

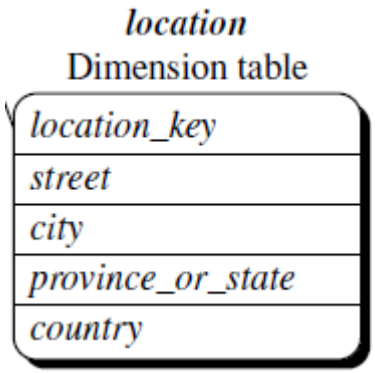
- Pivoting
  - It rotates the data axes in view in order to provide an alternative presentation of data.

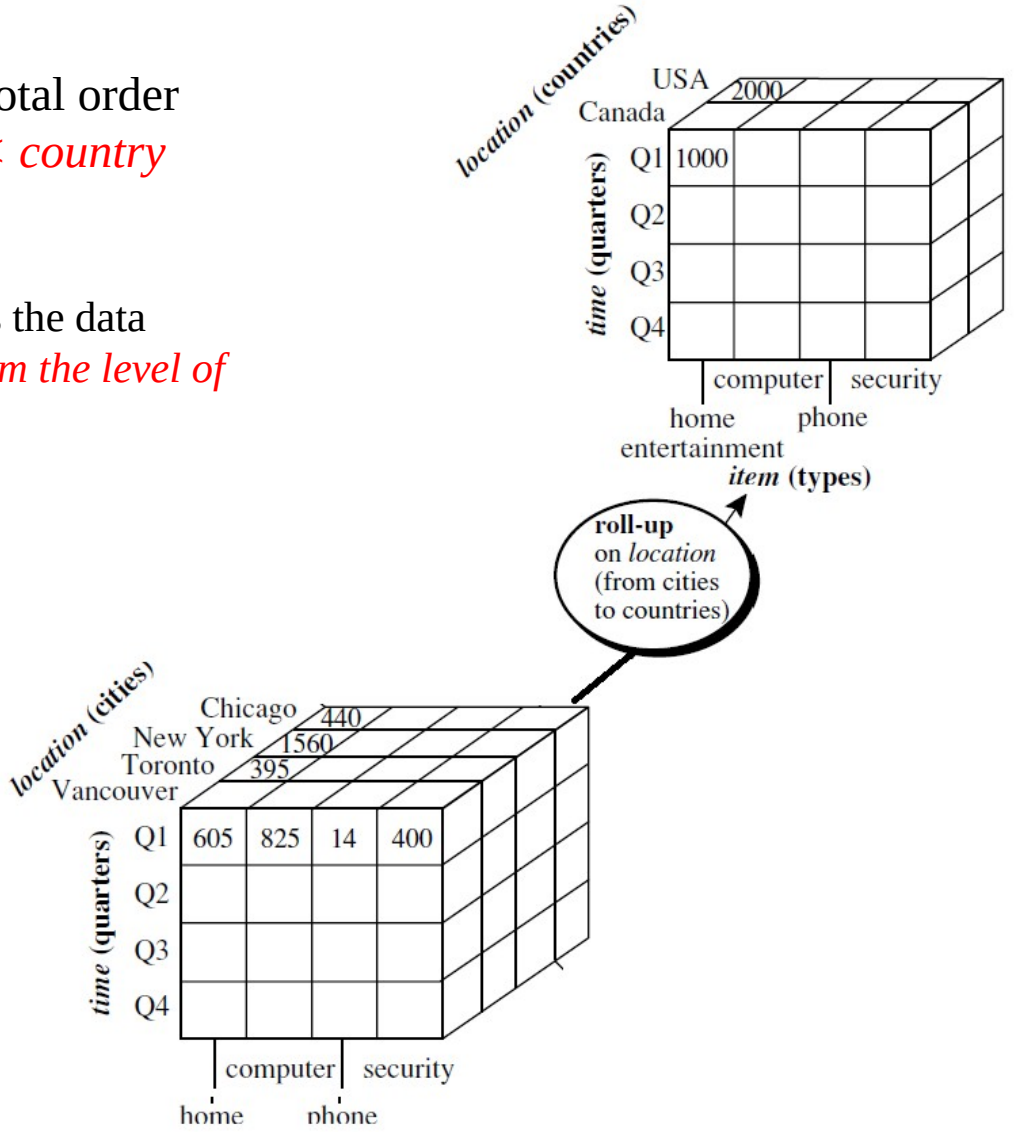


- Roll- up
  - The roll-up operation performs aggregation on a data cube
  - Example: The result of a roll-up Operation performed on the central cube by climbing up the concept hierarchy for *location*

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*

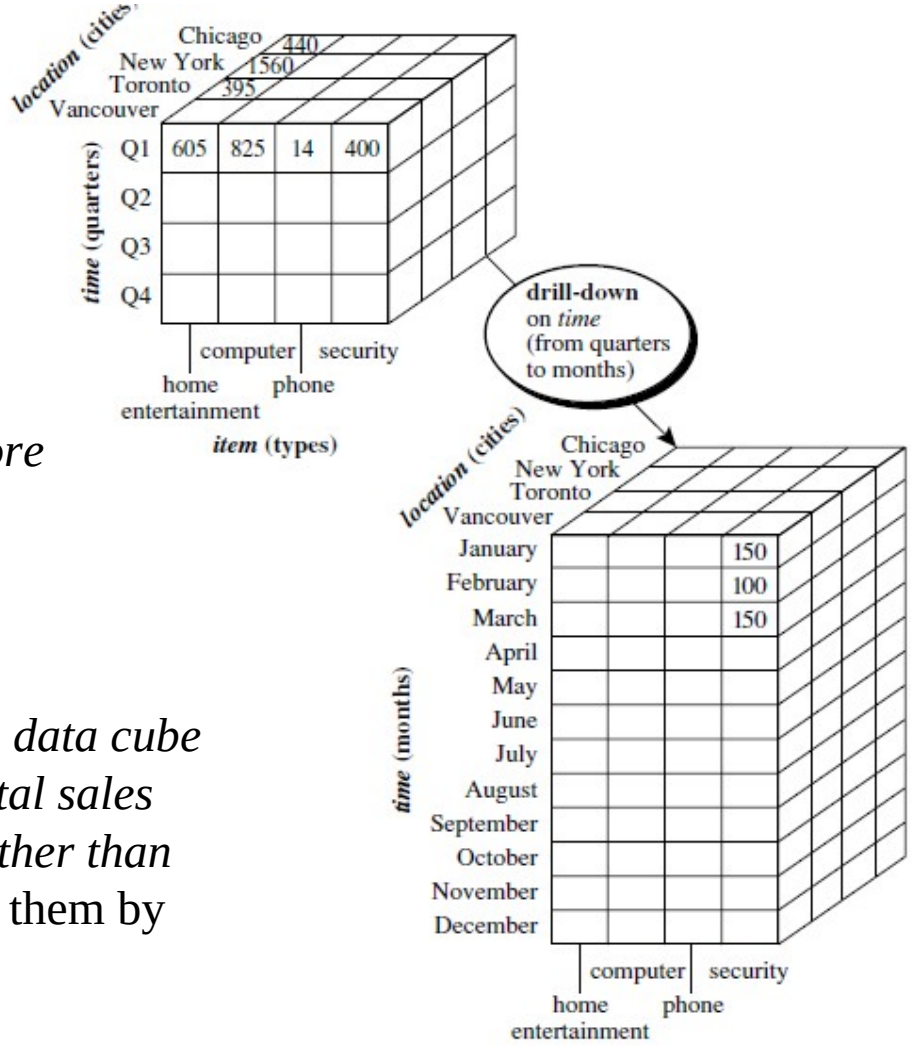




- **Drill-down** is the reverse of roll-up. It navigates from less detailed data to more detailed data.
- Drill-down can be realized by either *stepping down a concept hierarchy for a dimension or introducing additional dimensions*

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.*



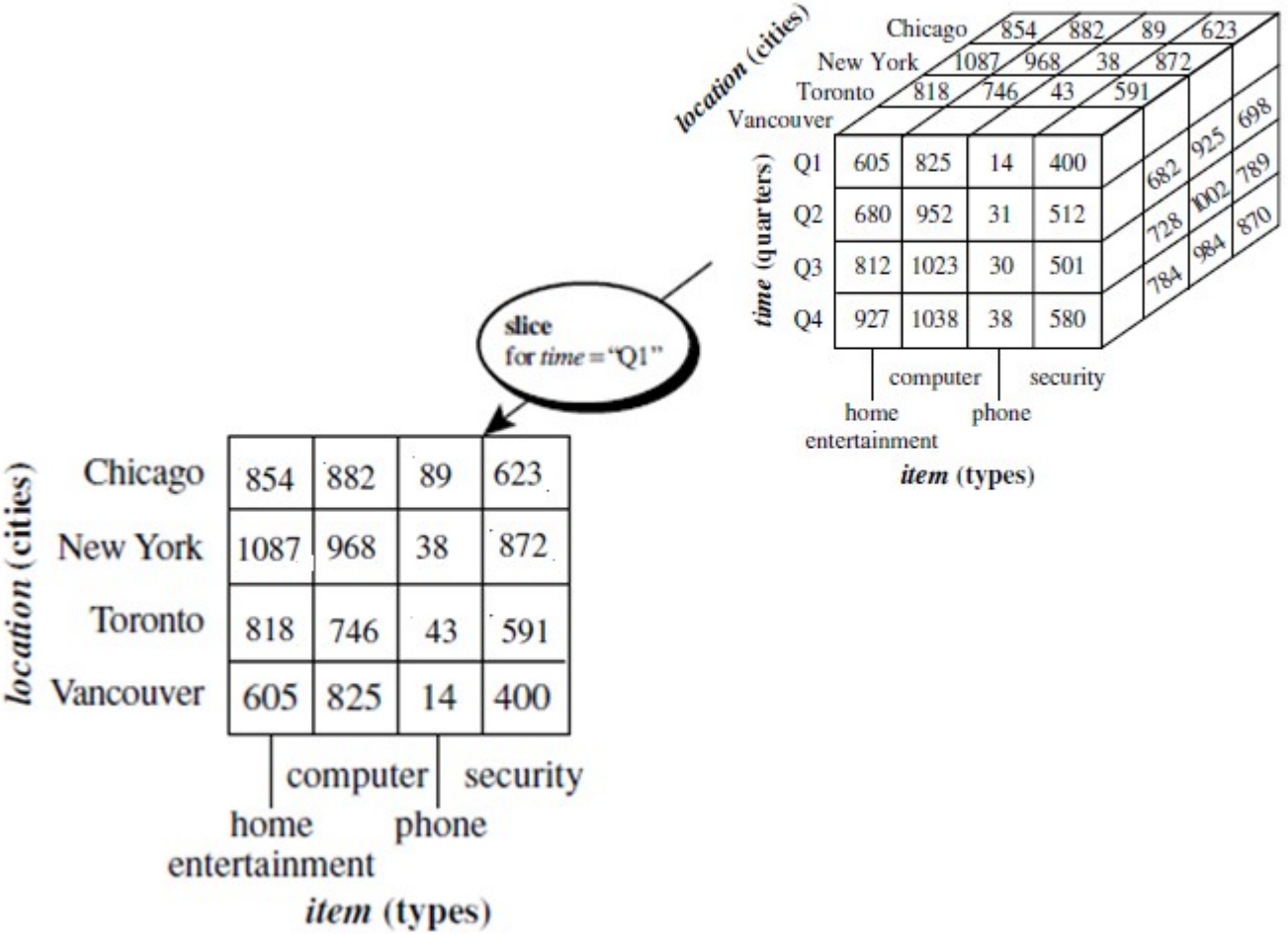
time
Dimension table

time_key
day
day_of_the_week
month
quarter
year

The resulting data cube details the total sales per month rather than summarizing them by quarter

- The ***slice*** *operation performs a selection on one dimension of the given* cube

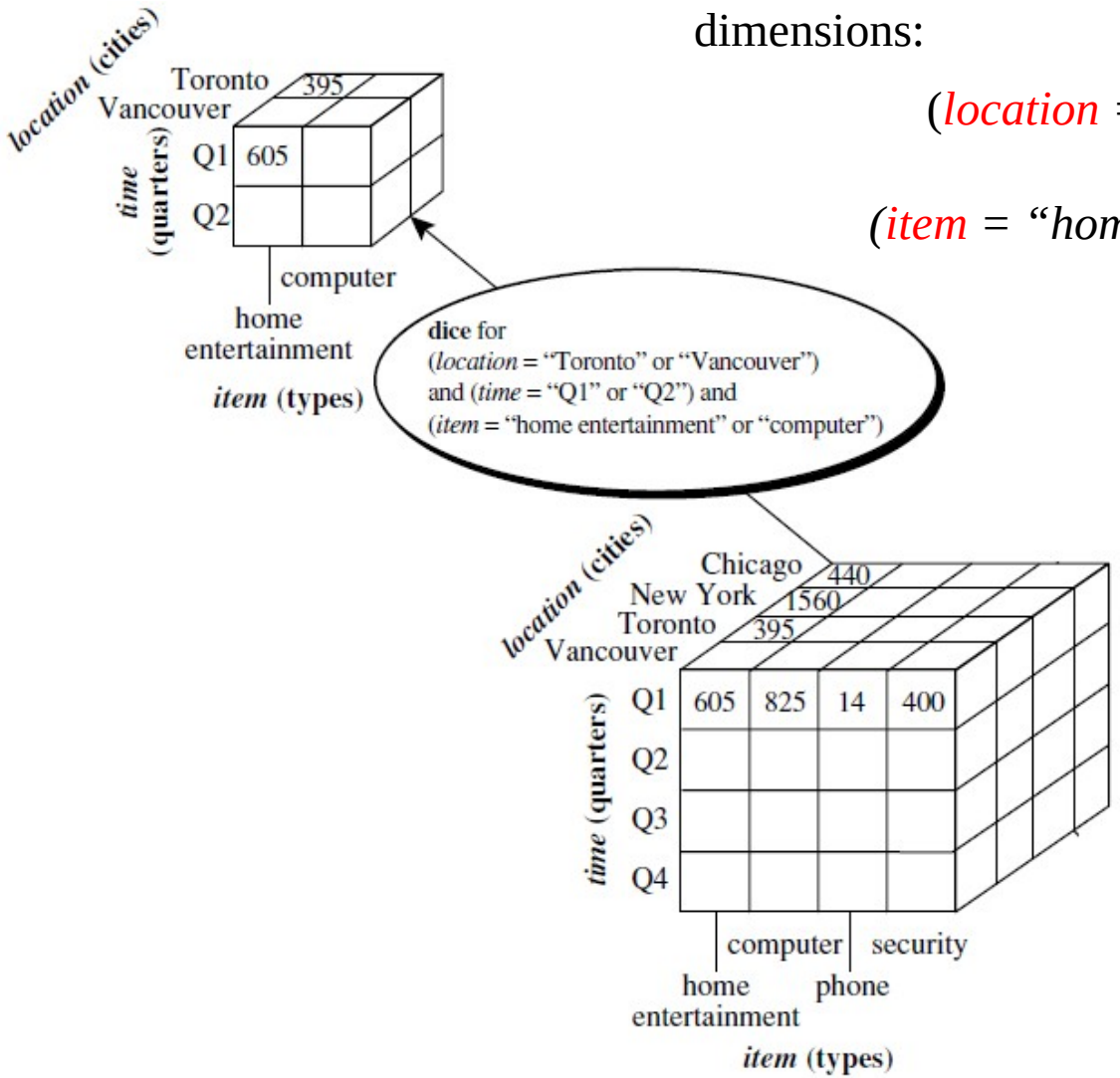
  The sales data are selected from the central cube for the dimension *time* *using the criterion time=“Q1.”*



- The *dice operation defines a subcube by performing a selection on two* or more dimensions

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 (also called rotate) is a visualization operation that rotates the data* axes in view to provide an alternative data presentation.

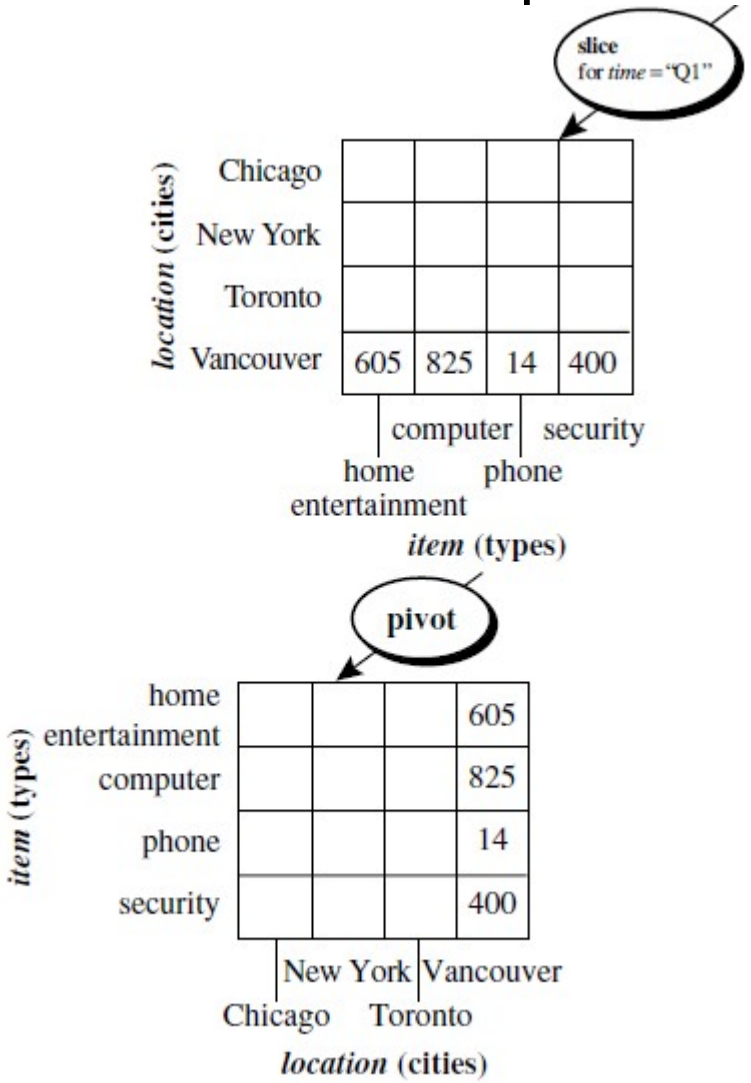
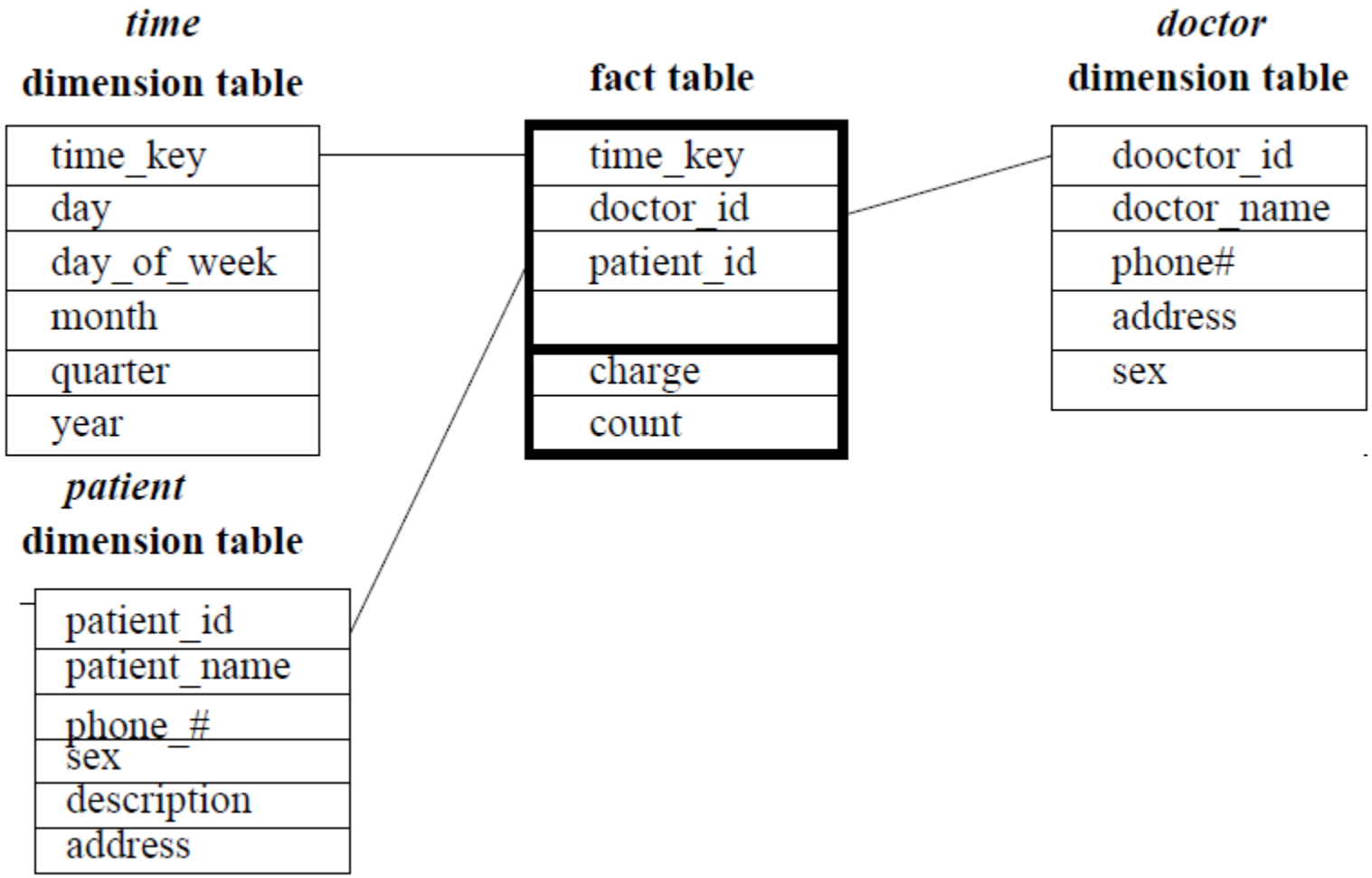


Figure shows a pivot operation where the *item and location axes in a 2-D slice are rotated*

- Suppose that a data warehouse consists of the three dimensions *time, doctor, and patient, and the two* measures *count and charge, where charge is the fee that a doctor charges a patient for a visit.*

  1. Draw as tar schema diagram for the above data warehouse
  2. Starting with the base cuboid [*day; doctor; patient], what specific OLAP operations should be per*formed in order to list the total fee collected by doctors in 2004?
  3. To obtain the same list, write an SQL query assuming the data is stored in a relational database with the schema *fee (day, month, year, doctor, hospital, patient, count, charge).*

a) Solution

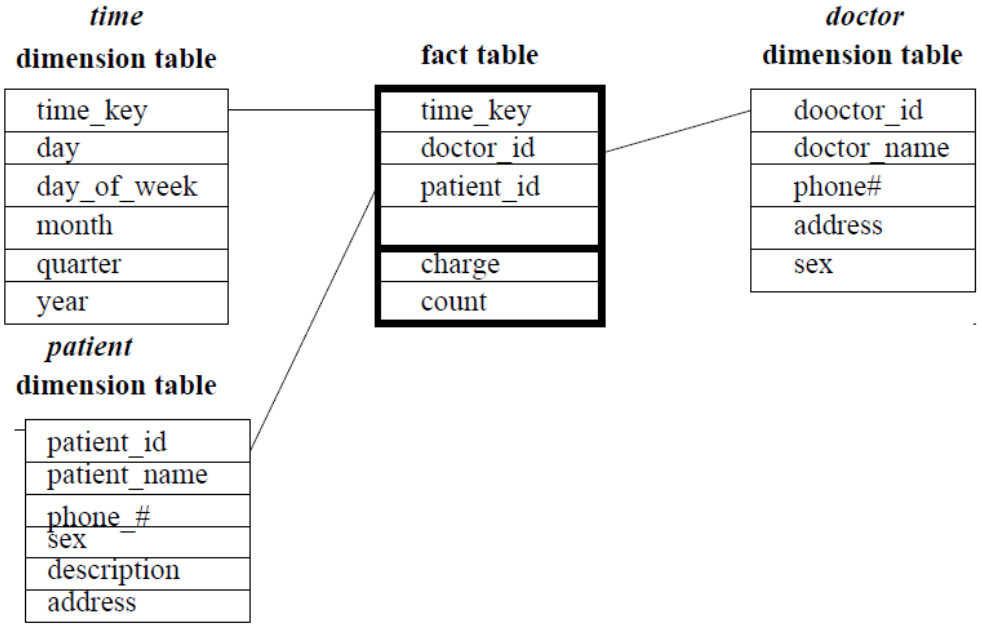

time
dimension table
time_key
day
day_of_week
month
quarter
year
fact table
time_key
doctor_id
patient_id
charge
count
doctor
dimension table
dooctor_id
doctor_name
phone#
address
sex
patient
dimension table
patient_id
patient_name
phone_#
sex
description
address

Starting with the base cuboid [*day; doctor; patient], what specific OLAP operations should be per*formed in order to list the total ***fee*** collected by doctors in 2004 ?

We need to give total fee collected in 2004 from patients (all)

fee : measure ; year : time

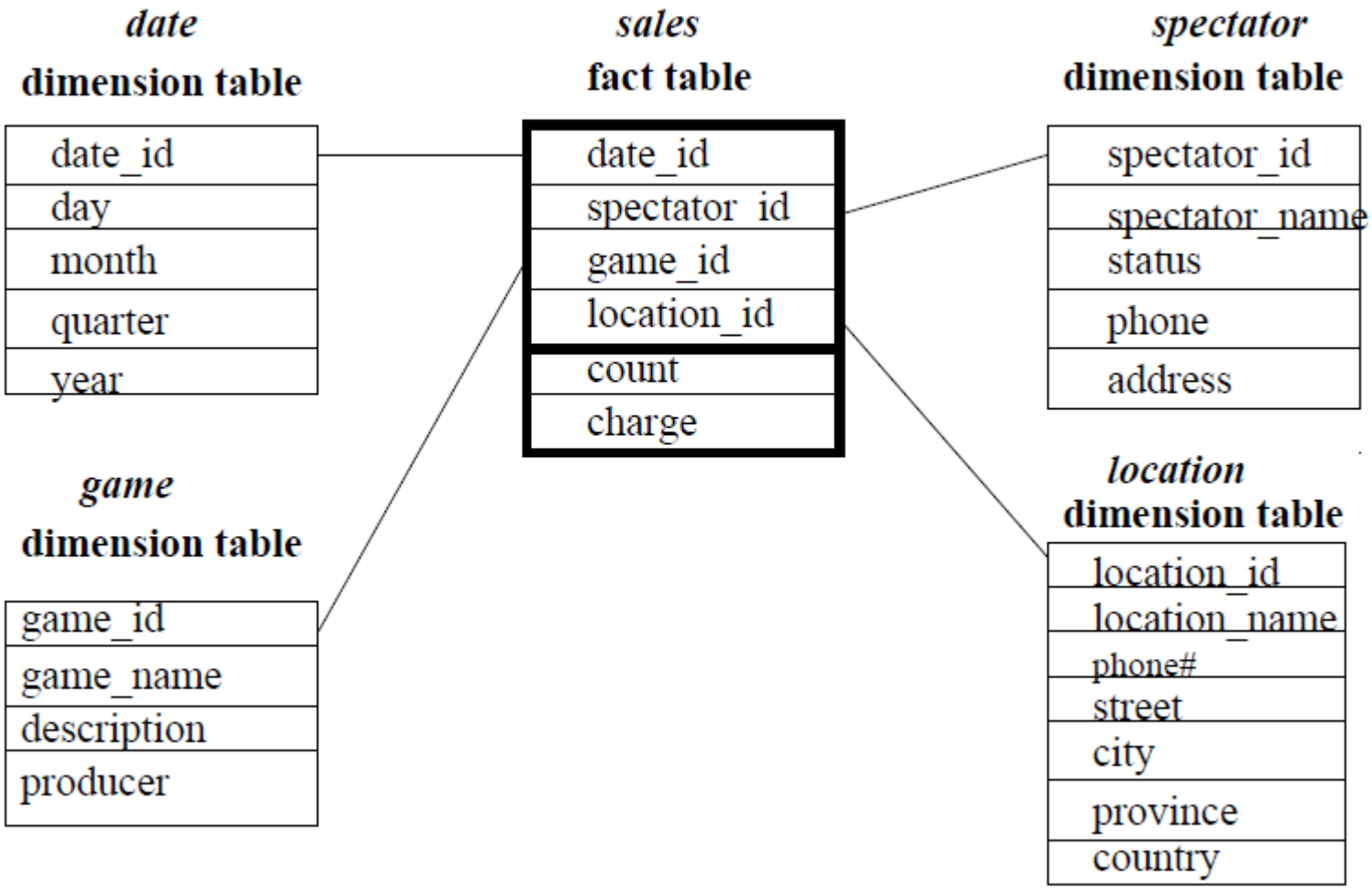
- Roll-up *on time from day to year.*
- Slice for *time=2004.*
- Roll-up on *patient from individual patient to all.*



- Suppose that a data warehouse consists of the four dimensions, *date, spectator, location, and game*, and the two measures, *count and charge*, *where charge is the fare that a spectator pays when watching a game on a* given date. *Spectators may be students, adults, or seniors*, with each category having its own charge rate.

(a) Draw a *star schema diagram for the data warehouse.*

(b)Starting with the base cuboid [*date; spectator; location; game], what specific OLAP operations should* one perform in order to list the total charge paid by student spectators at GM Place in 2004?

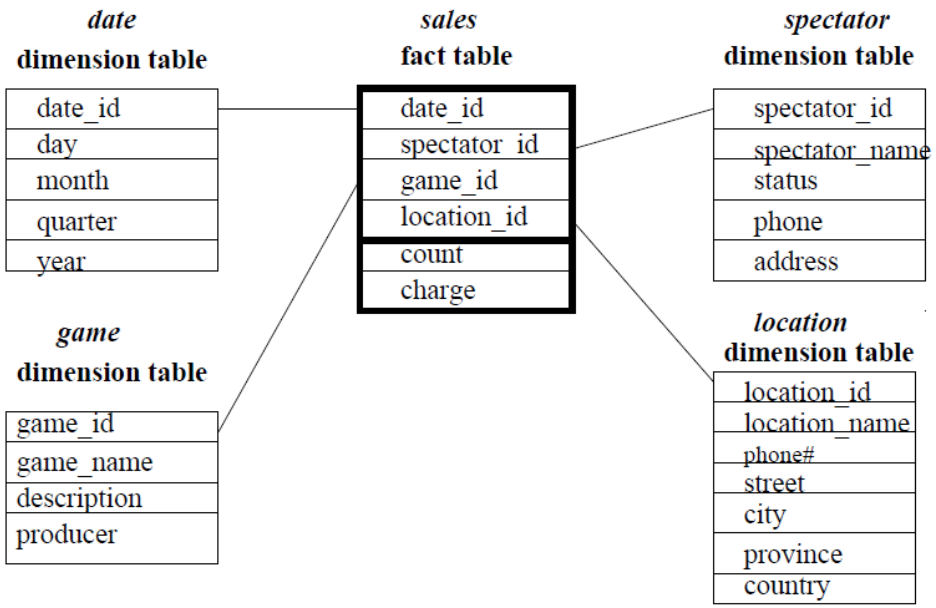

date
dimension table
date_id
day
month
quarter
year
sales
fact table
date_id
spectator_id
game_id
location_id
count
charge
spectator
dimension table
spectator_id
spectator_name
status
phone
address
game
dimension table
game_id
game_name
description
producer
location
dimension table
location_id
location_name
phone#
street
city
province
country

(b)Starting with the base cuboid [*date; spectator; location; game*], *what specific OLAP operations should* one perform in order to list the total *charge* paid by student spectators at GM Place in 2004?

charge – measure

Student - *Spectator*
GM Place - *Location*
2004 – *Date*
For all the games – Game

Spectator, Location, Date, Game are dimensions



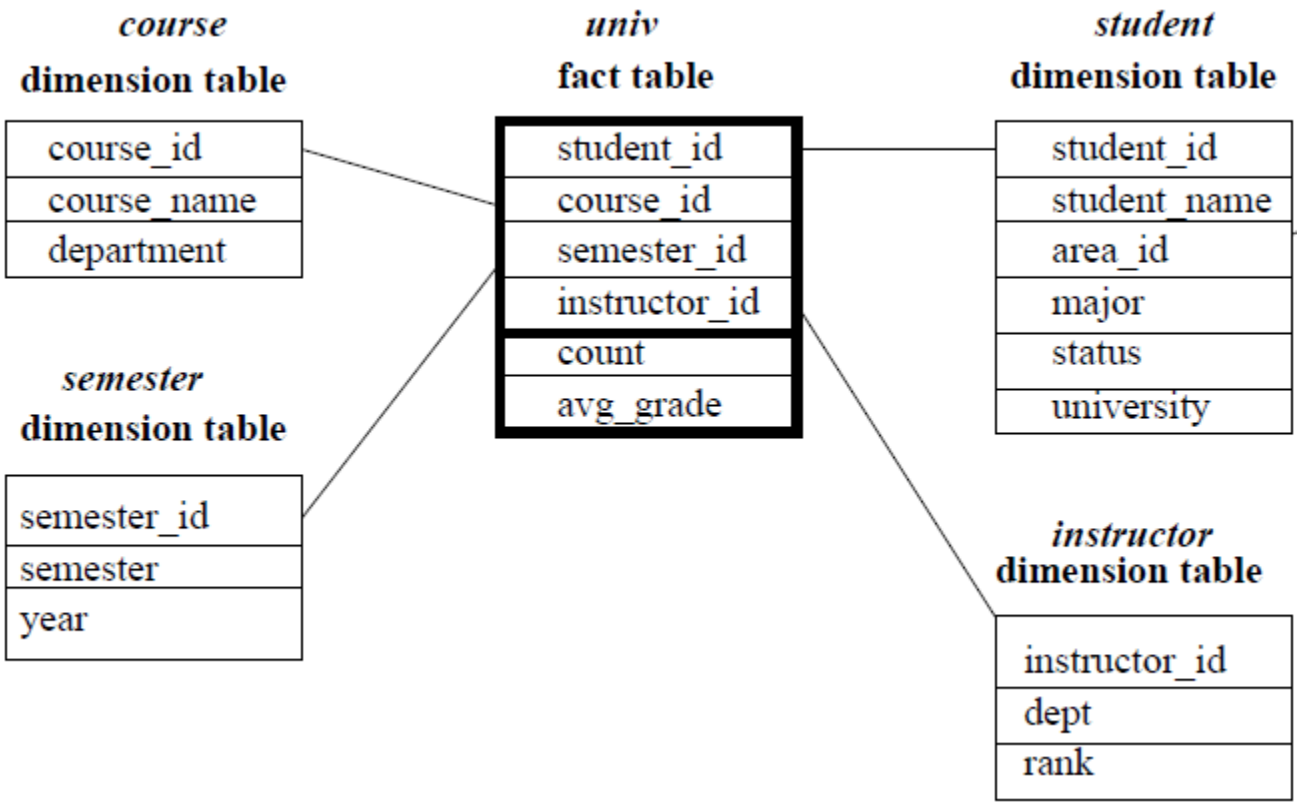
- **Roll-up** on *date from date_id to year.*
- **Roll-up** *on location from location_id to location_name.*
- **Roll-up** *on spectator from spectator id to status.*
- **Dice** *with status=“students", location_name=“GM Place” and year=2004.*
- **Roll-up** *on game from game_id to all.*

- Suppose that a data warehouse for *Big University consists of the following four dimensions: student, course, semester, and instructor, and two measures count and avg grade. When at the lowest conceptual level (e.g.,* for a given student, course, semester, and instructor Combination), the *avg grade measure stores the actual* course grade of the student. At higher conceptual levels, *avg grade stores the average grade for the given* combination.

(a) Draw a *snow-flake schema diagram for the data warehouse.*

(b) Starting with the base cuboid [*student; course; semester; instructor], what specific OLAP operations* (e.g., roll-up from *semester to year) should one perform in order to list the average grade of CS courses* for each *Big University student.*

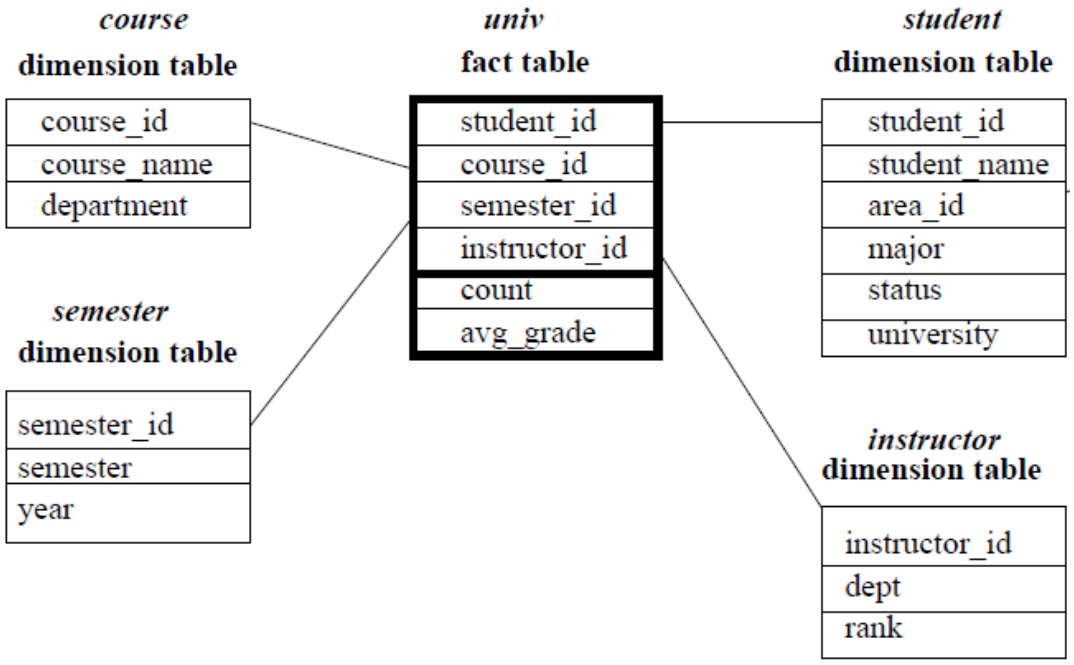
(c) If each dimension has ¯ve levels (including all), such as *student < major < status < university < all",* how many cuboids will this cube contain (including the base and apex cuboids)?


course
dimension table
course_id
course_name
department
semester
dimension table
semester_id
semester
year
univ
fact table
student_id
course_id
semester_id
instructor_id
count
avg_grade
student
dimension table
student_id
student_name
area_id
major
status
university
instructor
dimension table
instructor_id
dept
rank

what specific OLAP operations should one perform in order to list the average grade of CS courses for each *Big University* *student.*

avg grade *is measure*

CS - COURSE
BIG University – STUDENT
For each student - STUDENT



what specific OLAP operations should one perform in order to list the average grade of CS courses for each *Big University student.*

- Roll-up on course from *course id to department.*
- *Roll-up on student from student id to university.*
- *Dice on course, student with department="CS" and university = "Big University".*
- *Drill-down on student from university to student name.*

(c) If each dimension has five levels (including all), such as *student* $<$ *major* $<$ *status* $<$ *university* $<$ all, how many cuboids will this cube contain (including the base and apex cuboids)?
This cube will contain $5^4 = 625$ cuboids.

Exercise: Data Warehouse design for a wholesale furniture company

Exercise

Wholesale furniture company

Design the data warehouse for a wholesale furniture company. The data warehouse has to allow to analyze the company's situation at least with respect to the Furniture, Customers and Time.
Moreover, the company needs to analyze:

- the furniture with respect to its type (chair, table, wardrobe, cabinet...), category (kitchen, living room, bedroom, bathroom, office...) and material (wood, marble...)
- the customers with respect to their spatial location, by considering at least cities, regions and states

The company is interested in learning at least the quantity, income and discount of its sales.

Questions

1. Identify facts, dimensions and measures
2. For each fact:
   - design the star or snowflake schema and write the following SQL queries:
     - Find the quantity, the total income and discount with respect to each city, type of furniture and the month
     - Find the average quantity, income and discount with respect to each country, furniture material and year
     - Determine the 5 most sold furnitures during the May month

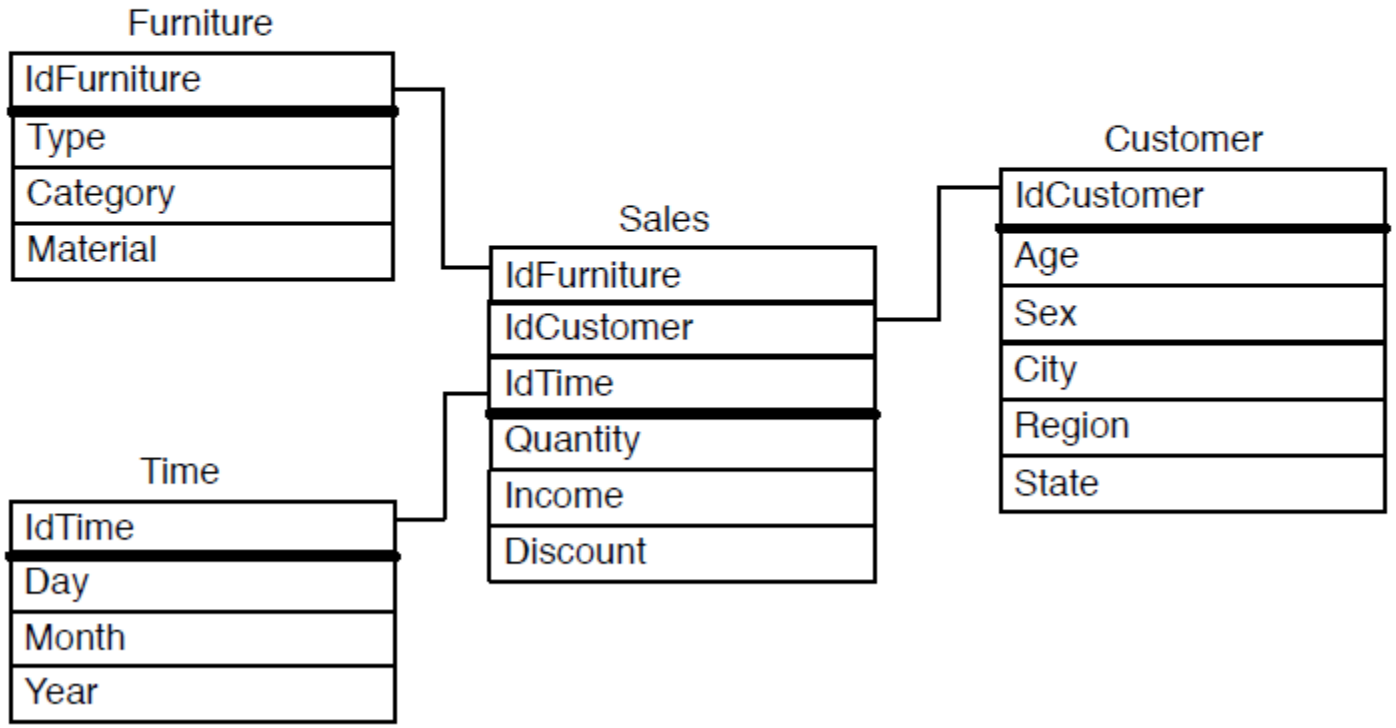
A possible solution

Facts, dimensions, measures, attribute tree, fact schema

FACT	Sales
MEASURES	Quantity, Income, Discount
DIMENSIONS	Furniture (Type, Category, Material)
	Customer (Age, Sex, City $\rightarrow$ Region $\rightarrow$ State)
	Time (Day $\rightarrow$ Month $\rightarrow$ Year)

A possible solution

Star schema



- Find the quantity, the total income and discount with respect to each city, type of furniture and the month

```
SELECT C.City, F.Type, T.Month,
           SUM(S.Quantity), SUM(S.Income), SUM(S.Discount)
FROM Sales S, Customer C, Time T, Furniture F
WHERE S.IdCustomer = C.IdCustomer AND
           S.IdTime = T.IdTime AND
           S.IdFurniture = F.IdFurniture
GROUP BY T.Month, F.Type, C.City
```

- Find the average quantity, income and discount with respect to each country, furniture material and year

```
SELECT C.Country, F.Material, T.Year,
           AVG(S.Quantity), AVG(S.Income), AVG(S.Discount)
FROM Sales S, Customer C, Time T, Furniture F
WHERE S.IdCustomer = C.IdCustomer AND
           S.IdTime = T.IdTime AND
           S.IdFurniture = F.IdFurniture
GROUP BY T.Year, C.Country, F.Material
```

- Determine the 5 most sold furnitures during the May month

```
SELECT F.Type, SUM(S.Quantity)
FROM (
   SELECT F.Type, SUM(S.Quantity) AS TotQuantity,
      RANK() OVER (ORDER BY SUM(S.Quantity) DESC)
                              AS Rank
   FROM Sale S, Furniture F, Time T
   WHERE S.IdFurniture = F.IdFurniture AND
            S.IdTime = T.IdTime AND
            T.Month = "May")
WHERE rank <=5
```

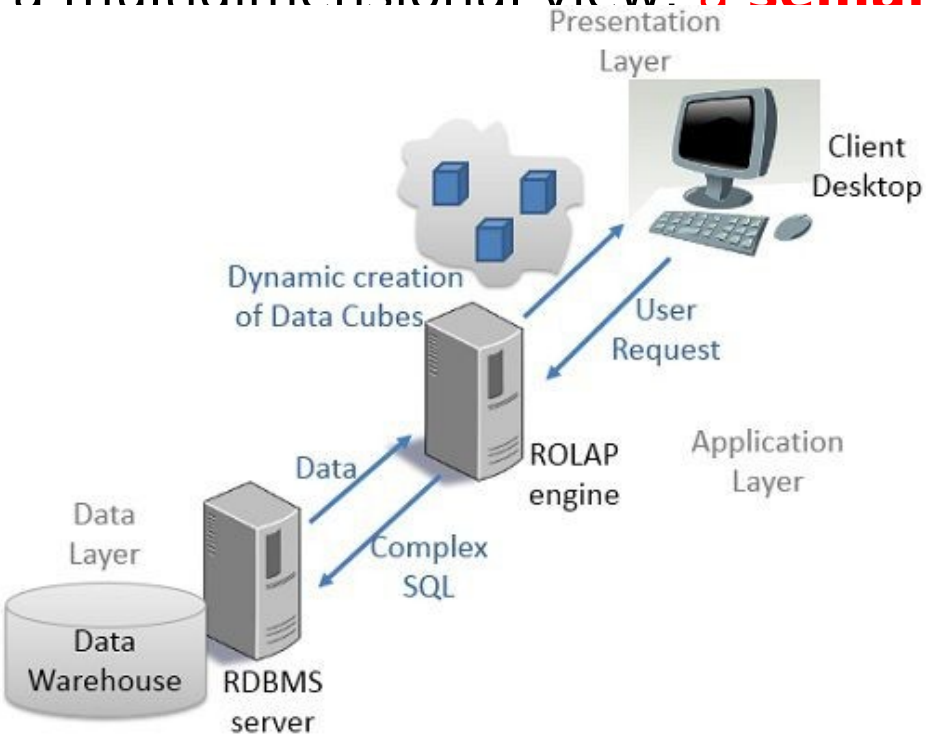
OLAP Server Architectures: ROLAP versus MOLAP versus HOLAP

- Data cube products use different techniques for pre-computing aggregates and storing them.
- They are generally based on one of two implementation models.

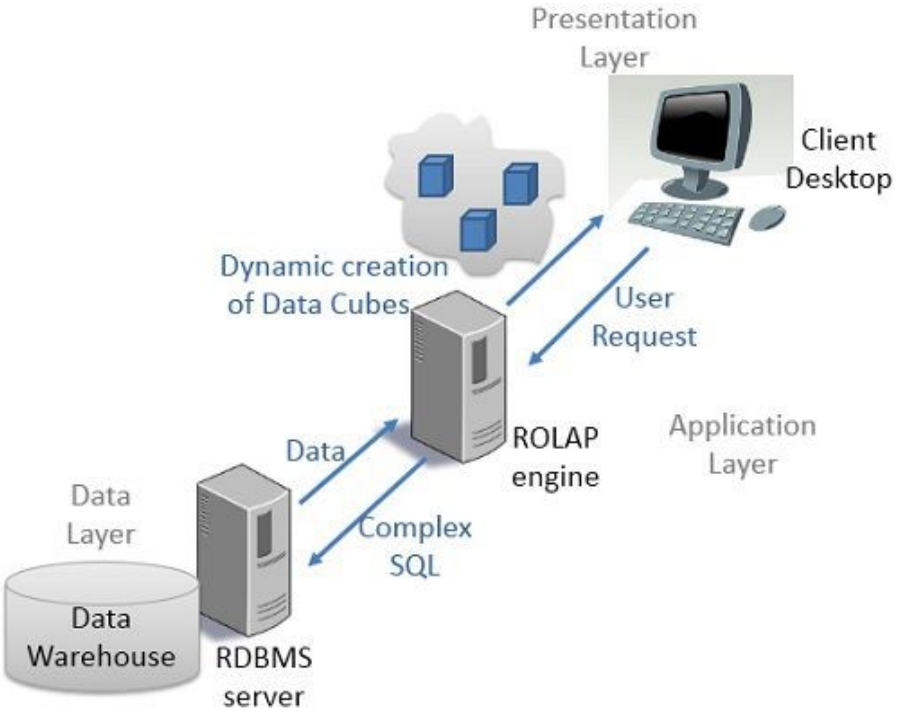
1. The ROLAP model or the Relational OLAP model
2. The MOLAP model for multidimensional OLAP.

Hybrid OLAP (HOLAP) refers to technologies that combine MOLAP and ROLAP.

- **ROLAP** is **Relational Online Analytical Processing** model,
  - The data is stored as in relational database i.e. **rows and columns** in the data warehouse.
  - Data is presented in front of the user in the **multidimensional** form.
  - To display the data, in a multidimensional view, a **semantic layer of metadata** is the relational tables.



ROLAP Model



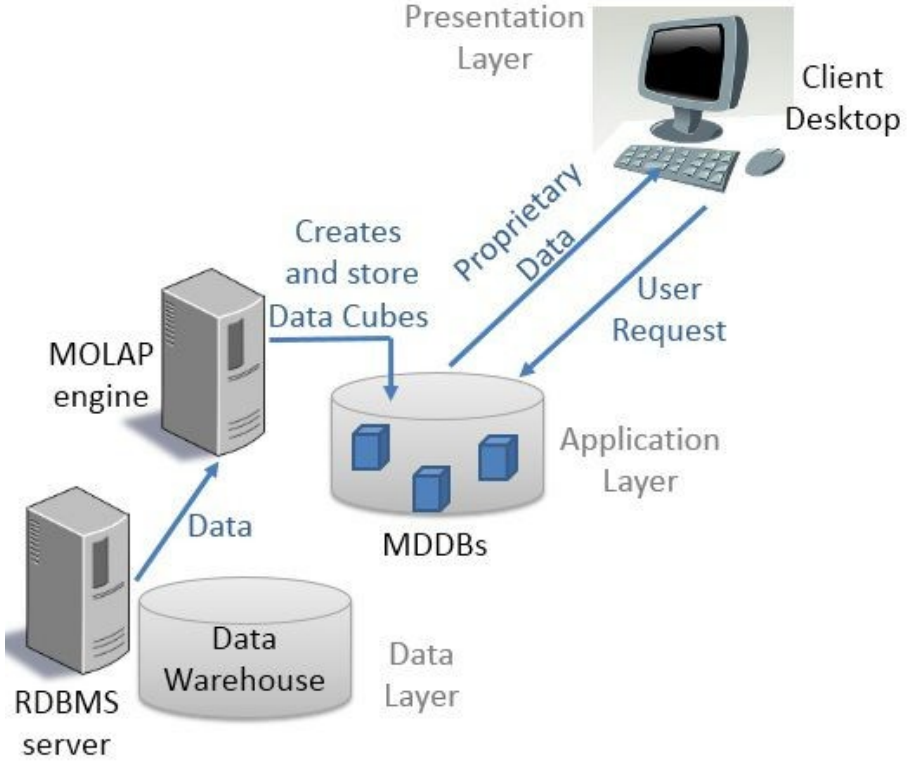
ROLAP Model

- Whenever the ROLAP engine issues a complex query, it fetches data from the main warehouse and **dynamically** creates a multidimensional view of data for the user.
- As the multidimensional view of data is created dynamically it processes **slower** in comparison to MOLAP. ROLAP engine deals with **large volumes** of data.

OLAP Server Architectures: ROLAP

- The **advantages,** It can handle a large amount of data and can leverage all the functionalities of the relational database
- The **disadvantages,** the performance is slow and each ROLAP report is an SQL query. It is difficult to perform complex calculations using SQL
- It is also limited by SQL functionalities.

- **MOLAP** is a **Multidimensional Online Analytical Processing** model.
  - The data used for analysis is stored in specialized **multidimensional databases (MDDBs)**.
  - The multidimensional database management systems are **proprietary software systems**.
  - These multidimensional databases are formed from the large multidimensional **array**.
  - The cells or data cubes of this multidimensional databases carry **precalculated** and **prefabricated** data.
  - Proprietary software systems create this precalculated and fabricated data, while the data is loaded to MDDBs from the main databases.



MOLAP Model

- Now, it is the work of MOLAP engine, which reside there in the application layer, provide the multidimensional view of data from MDDBs to the user.
- Thus when a user request for the data, no time is wasted in calculating the data and the system responses fast.

OLAP Server Architectures : MOLAP

- MOLAP is the more traditional way of OLAP analysis. In MOLAP, data is stored in a multidimensional cube. The storage is not in the relational database, but in proprietary formats.
  - **Advantages:**
    - Excellent performance: MOLAP cubes are built for fast data retrieval, and are optimal for slicing and dicing operations.
    - Can perform complex calculations: All calculations have been pre-generated when the cube is created. Hence, complex calculations are not only doable, but they return quickly.

OLAP Server Architectures : MOLAP

- Disadvantages:
  - **Limited in the amount of data it can handle:** Because all calculations are performed when the cube is built, it is not possible to include a large amount of data in the cube itself.
  - **Requires additional investment**: Cube technology are often proprietary and do not already exist in the organization. Therefore, to adopt MOLAP technology, chances are additional investments in human and capital resources are needed.

OLAP Server Architectures : HOLAP

- **HOLAP**
  - HOLAP technologies attempt to combine the advantages of MOLAP and ROLAP.
  - For summary-type information, HOLAP leverages cube technology for faster performance.
  - When detail information is needed, HOLAP can "drill through" from the cube into the underlying relational data.

BASIS FOR COMPARISON	ROLAP	MOLAP
Storage & Fetched	Data is stored and fetched from the main data warehouse.	Data is Stored and fetched from the Proprietary database MDDBs.
Data Form	Data is stored in the form of relational tables.	Data is Stored in the large multidimensional array made of data cubes.
Data volumes	Large data volumes.	Limited summaries data is kept in MDDBs.
Technology	Uses Complex SQL queries to fetch data from the main warehouse.	MOLAP engine created a precalculated and prefabricated data cubes for multidimensional data views.
View	ROLAP creates a multidimensional view of data dynamically.	MOLAP already stores the static multidimensional view of data in MDDBs.
Access	Slow access.	Faster access.