Data Warehouse

——A Multi-dimensional data model——

徐华

清华大学 计算机系 智能技术与系统国家重点实验室

xuhua@tsinghua.edu.cn

1

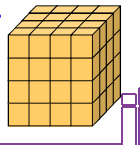
Data Model

- Review the basic concepts of database
- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- From data warehousing to data mining

2

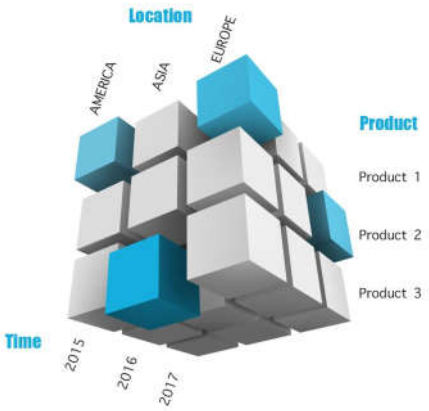
Data Cube (1)

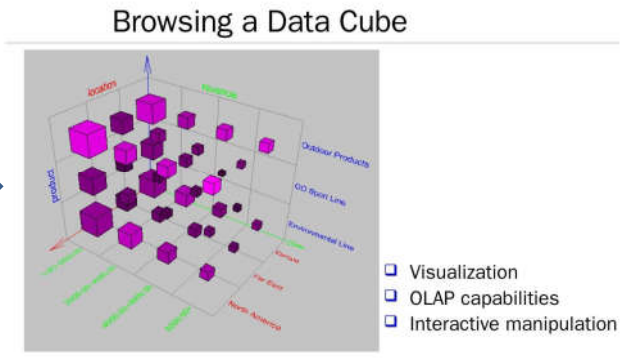
- A data warehouse is based on a multidimensional data model which views data in the form of a data cube
- A data cube, such as sales, allows data to be modeled and viewed in multiple dimensions
  - Dimension tables (维表) , such as item (item_name, brand, type), or time(day, week, month, quarter, year)
  - Fact table （事实表） contains measures (such as dollars_sold) and keys to each of the related dimension tables
- In data warehousing literature, an n-D base cube is called a base cuboid(基本方体). The top most 0-D cuboid, which holds the highest-level of summarization, is called the apex cuboid(顶端方体). The lattice of cuboids forms a data cube.



3

Data Cube (2)





31

4

Data Cube (3)

- Dimension and Dimension table
  - Dimension: is the perspectives or entities with respect to which an organization wants to keep records.
  - Dimension table: is a set of properties to further describes a dimension.
- Each dimension may be associated with a dimension table.

  Time, item, location, provider
- Fact and fact table
  - Fact: the measure of a theme
  - Fact table: the representation of the fact. It contains the names of the facts, keys to each of the related dimension tables. Facts are numerical, sales amount

5

Data Cube (4)

- Dimension number of data cube

  The number of dimensions to be viewed.

  Sales (item time location dollars_sold)

  Base cuboid: the cube which contains all dimensions that can be viewed in data warehousing.

  Apex cuboid: the cube which contain no dimension.

  Data Cube: is the all cuboids in a multi dimensional data model.

6

Data Cube — One Example(1)

ALLElectronics sales

dimension : time , item , location , brand
dimension table :
time(time_key day day_of_week month quarter year)
item(item_key item_name brand type supplier_key)
fact table : (time_key item_key brand_key location_key dollars_sold units_sold)

7

Data Cube — One Example(2)

2-dimension data cube:

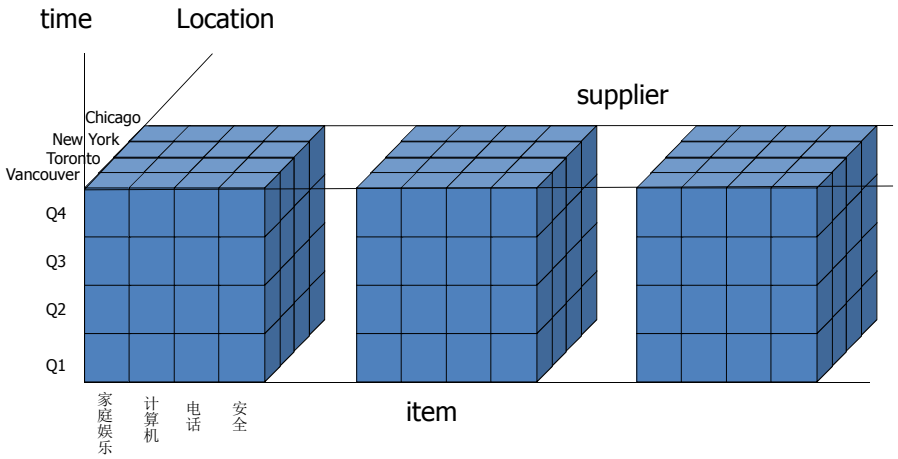
location＝” Vancouver”

	item(type)		
Time(quarter)	**entertainment**	computer	security
Q1	605	825	400
Q2	680	920	512
Q3	781	1026	501
Q4	824	1120	580

8

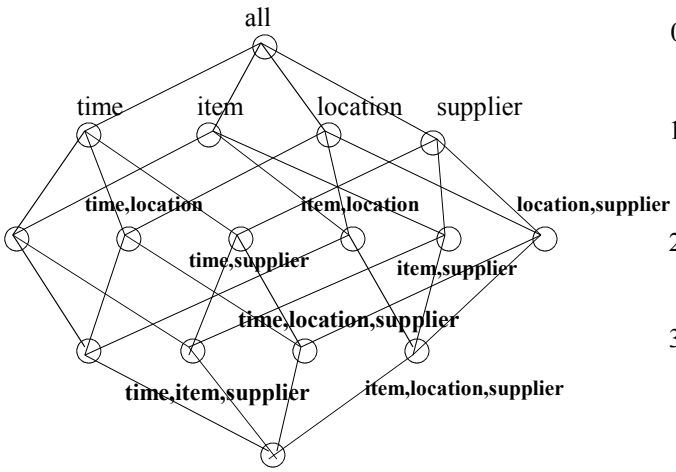
Data Cube — One Example(3)

- **4-dimension data cube**



9

Cube: A Lattice of Cuboids



0-D(apex) cuboid

1-D cuboids

2-D cuboids

3-D cuboids

4-D(base) cuboid

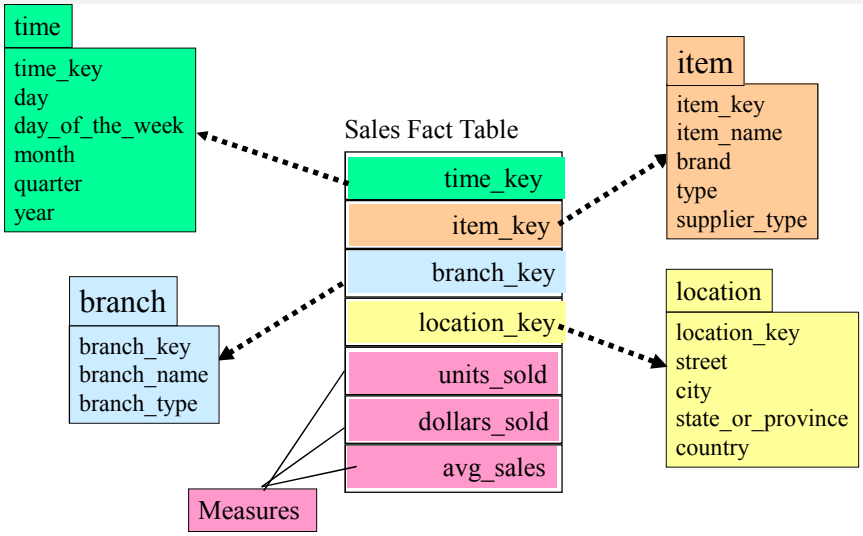
10

Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures
  - Star schema: A fact table in the middle connected to a set of dimension tables
  - Snowflake schema: A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake
  - Fact constellations (事实星座): Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation

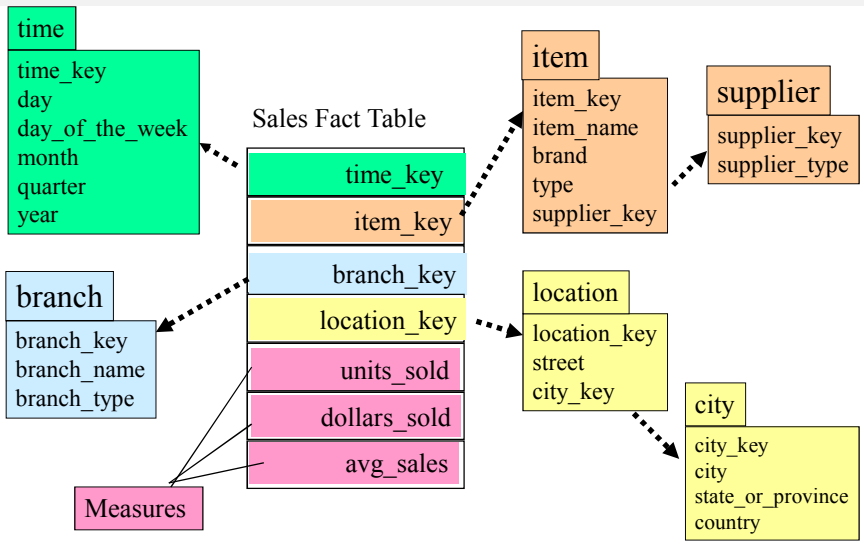
11

Example of Star Schema



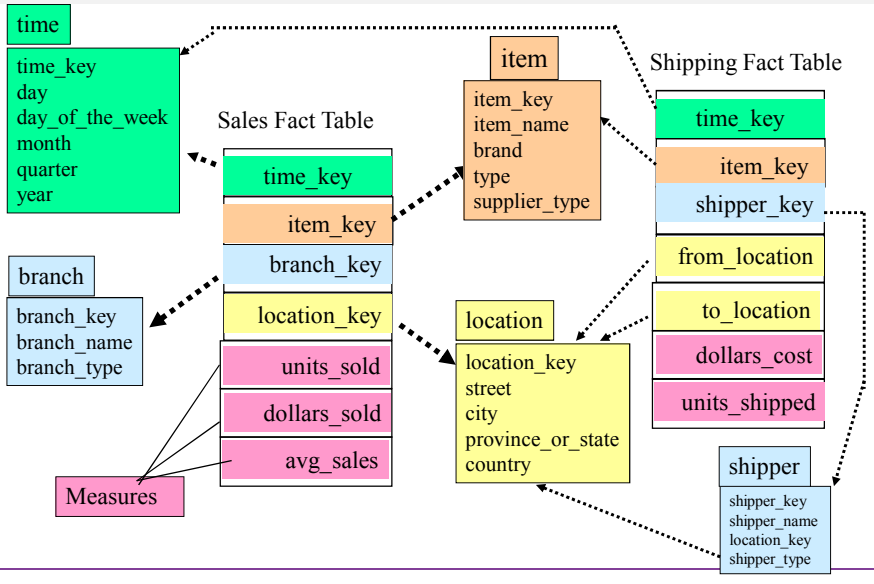
12

Example of Snowflake Schema



13

Example of Fact Constellations Schema



14

Cube Definition Syntax (BNF) in DMQL

- **Cube Definition (Fact Table)**
  define cube <cube_name> [<dimension_list>]: <measure_list>
- **Dimension Definition (Dimension Table)**
  define dimension <dimension_name> as (<attribute_or_subdimension_list>)
- **Special Case (Shared Dimension Tables)**
  - First time as "cube definition"
  - define dimension <dimension_name> as <dimension_name_first_time> in cube <cube_name_first_time>

15

Defining Star Schema in DMQL

define cube sales_star [time, item, branch, location]:

dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)

define dimension time as (time_key, day, day_of_week, month, quarter, year)

define dimension item as (item_key, item_name, brand, type, supplier_type)

define dimension branch as (branch_key, branch_name, branch_type)

define dimension location as (location_key, street, city, province_or_state, country)

16

Defining Snowflake Schema in DMQL

```
define cube sales_snowflake [time, item, branch, location]:
        dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)
define dimension time as (time_key, day, day_of_week, month, quarter, year)
define dimension item as (item_key, item_name, brand, type, supplier(supplier_key,
   supplier_type))
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as (location_key, street, city(city_key, province_or_state,
   country))
```

17

Defining Fact Constellation in DMQL

```
define cube sales [time, item, branch, location]:
        dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)
define dimension time as (time_key, day, day_of_week, month, quarter, year)
define dimension item as (item_key, item_name, brand, type, supplier_type)
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as (location_key, street, city, province_or_state, country)
define cube shipping [time, item, shipper, from_location, to_location]:
        dollar_cost = sum(cost_in_dollars), unit_shipped = count(*)
define dimension time as time in cube sales
define dimension item as item in cube sales
define dimension shipper as (shipper_key, shipper_name, location as location in cube sales,
   shipper_type)
define dimension from_location as location in cube sales
define dimension to_location as location in cube sales
```

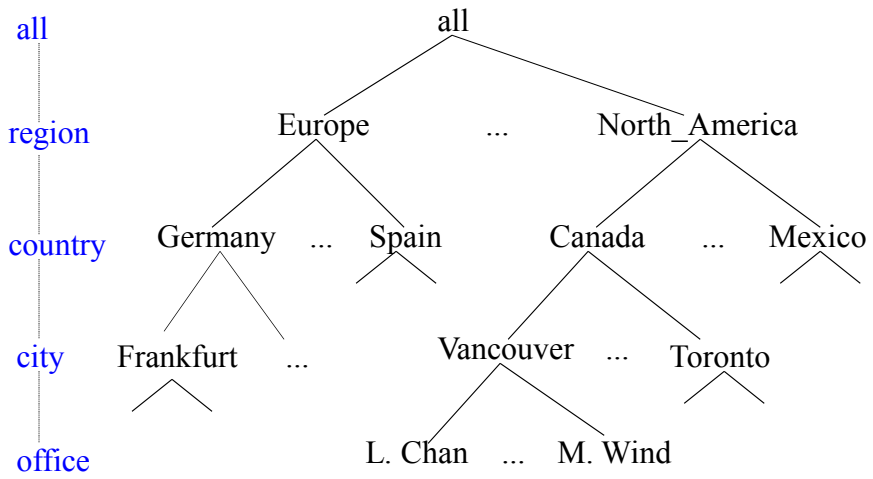
18

A Concept Hierarchy: Dimension

- A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts.
  - categories:
    - the hierarchy of property: location, province, country
    - the hierarchy or grouping of property value
- For a given dimension, there may be more than one concept hierarchy.

19

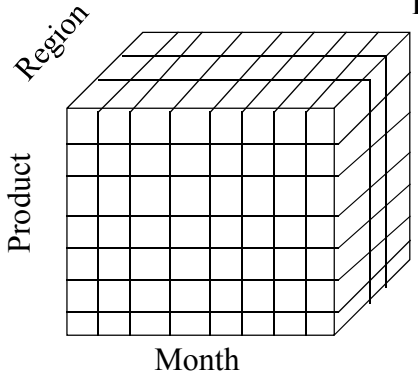
A Concept Hierarchy: Dimension (location)



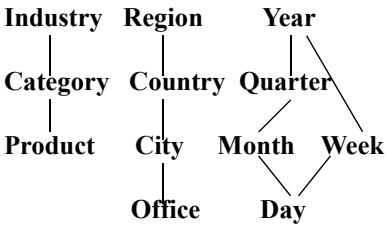
20

Multidimensional Data

- Sales volume as a function of product, month, and region

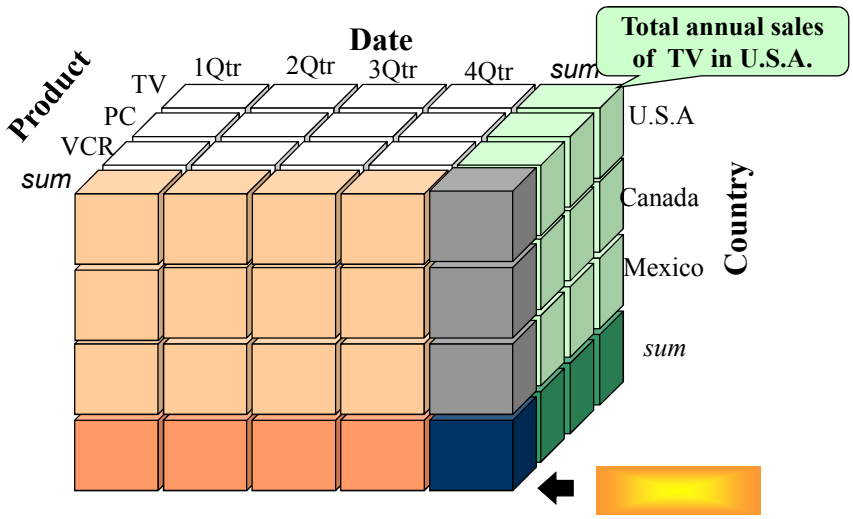


Dimensions: Product, Location, Time
Hierarchical summarization paths



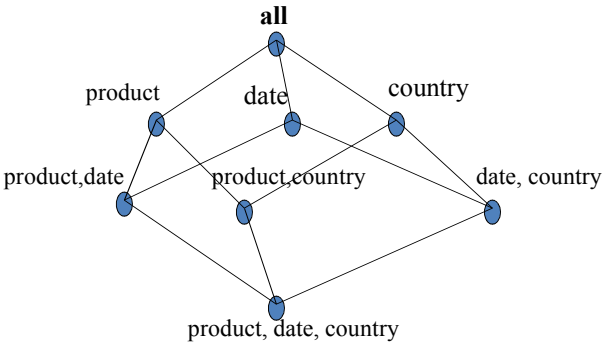
21

A Sample Data Cube



22

Cuboids Corresponding to the Cube



0-D(apex) cuboid

1-D cuboids

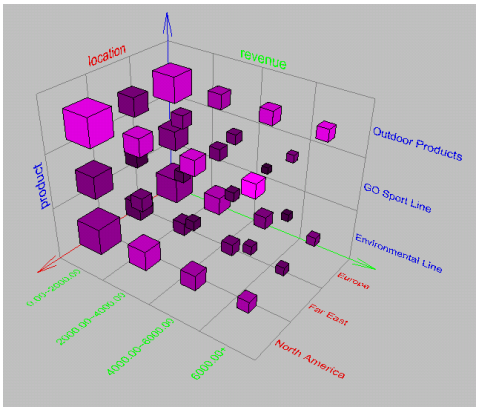
2-D cuboids

3-D(base) cuboid

23

View of Warehouses and Hierarchies

- **Visualization**
- **OLAP capabilities**
- **Interactive manipulation**



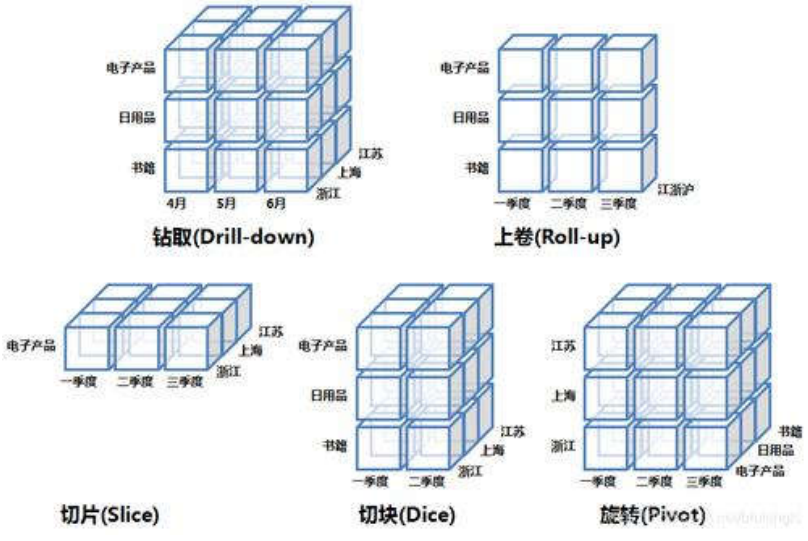
24

Typical OLAP Operations

- **Roll up (drill-up):** summarize data
  - *by climbing up hierarchy or by dimension reduction*
- **Drill down (roll down):** reverse of roll-up
  - *from higher level summary to lower level summary or detailed data, or introducing new dimensions*
- **Slice and dice:** *project and select on one or more dimensions*
- **Pivot (rotate):**
  - *reorient the cube, visualization, 3D to series of 2D planes*
- Other operations
  - *drill across: involving (across) more than one fact table*
  - *drill through: through the bottom level of the cube to its back-end relational tables (using SQL)*

25

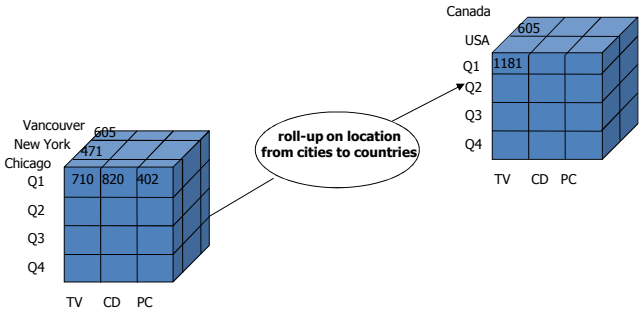
Typical OLAP Operations



26

Typical OLAP Operations (1)

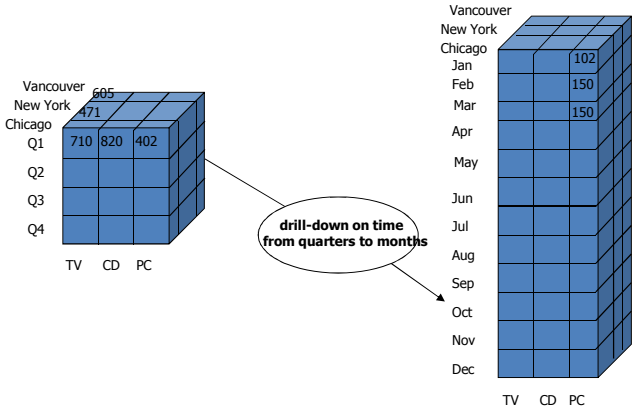
- **Roll up 上卷 (drill-up上钻):** summarize data
  - *by climbing up hierarchy (dimension reduction)*



27

Typical OLAP Operations(2)

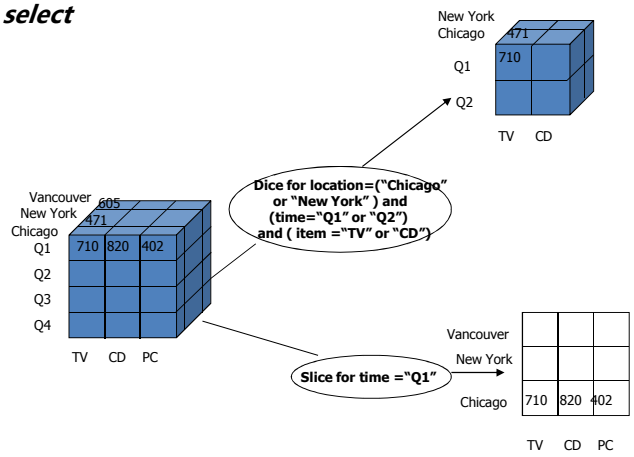
- **Roll down下卷 (Drill down下钻):** reverse of roll-up
  - *from higher level summary to lower level summary or detailed data, or introducing new dimensions*



28

Typical OLAP Operations(3)

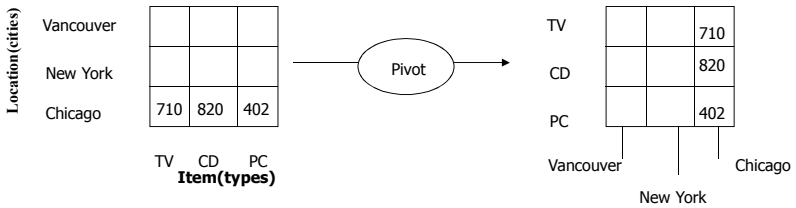
- **Slice(切片) and dice(切块):**
  - ***project and select***



29

Typical OLAP Operations (4)

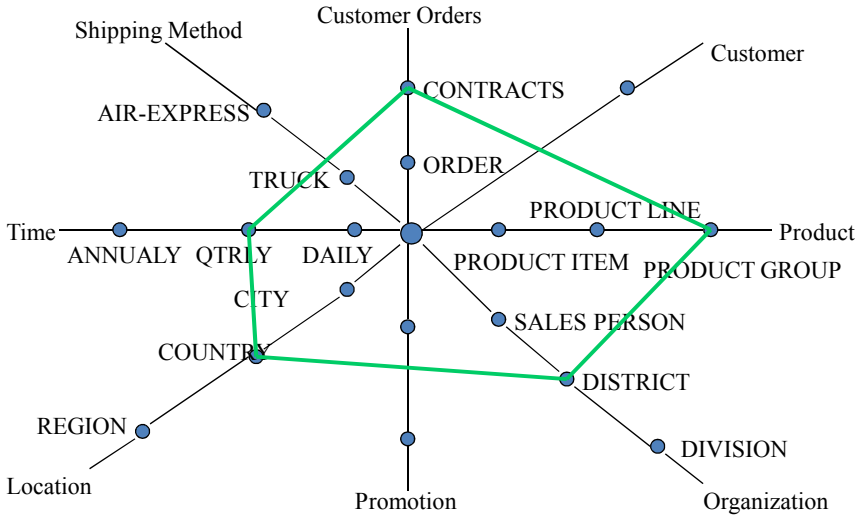
- **Pivot 旋转 (rotate旋转):**
  - ***reorient the cube, visualization, 3D to series of 2D planes.***



- **Other operations**
  - ***drill across:* involving (across) more than one fact table**
  - ***drill through:* through the bottom level of the cube to its back-end relational tables (using SQL)**

30

A Star-Net Query Model



Each circle is called a **footprint**.

31

Thanks !

32