Data Models for Data warehouse

Star Schema -

one fact table and number of dimension tables
all dimension table are connected with fact table
(centralized table)

Snowflake Schema

one fact table and number of dimension tables
dimensional table is partitioned into additional table
all dimension tables are connected with fact table

Fact Constellation Schema

more than one fact tables
dimension tables are connected with fact tables

Data Models for Data warehouse

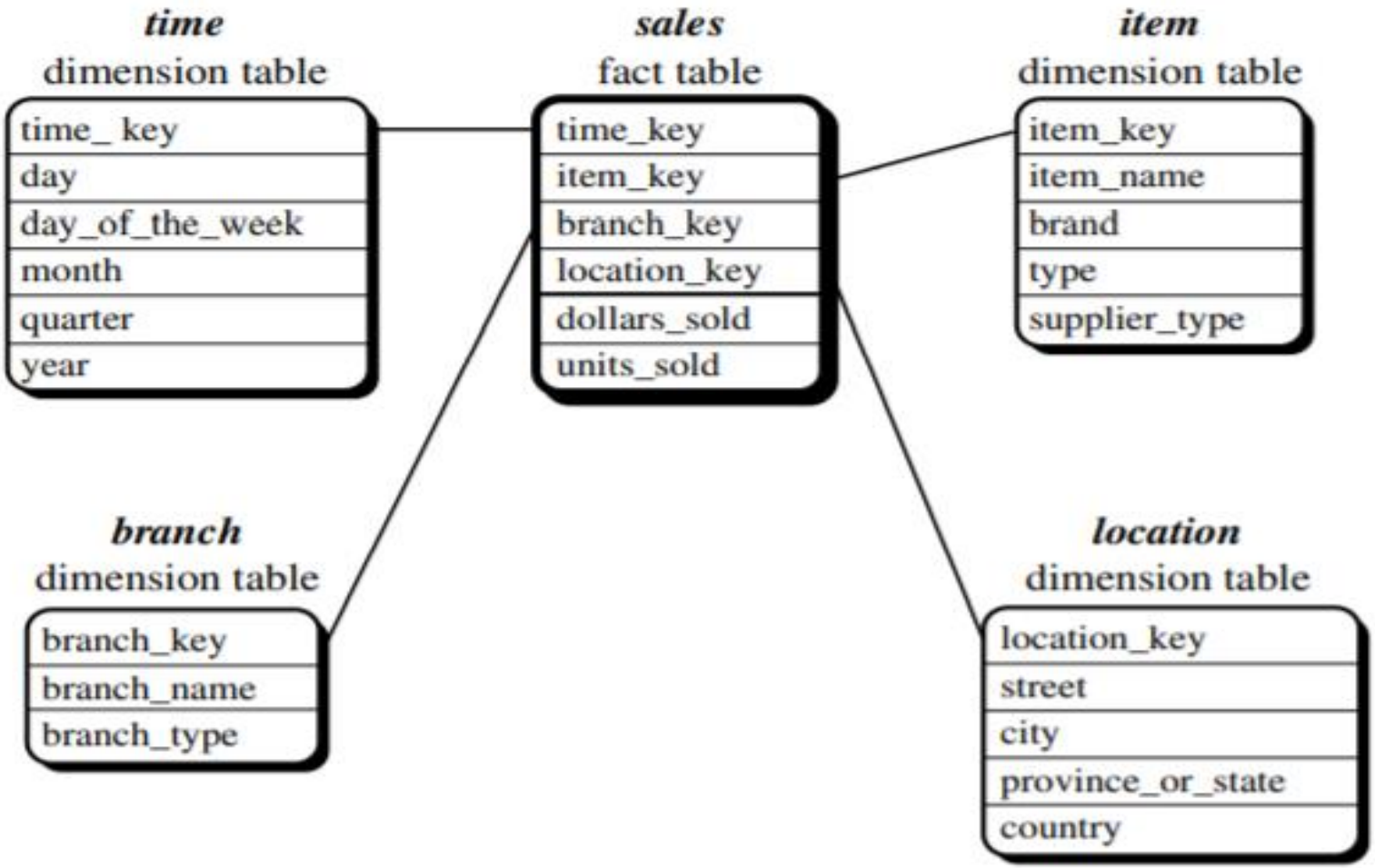
Star Schema

Sales are considered along four dimensions, namely, time, item, branch, and location.

The schema contains a central fact table for sales that contains keys to each of the four dimensions, along with two measures: dollars sold and units sold -

Data Models for Data warehouse

Star Schema



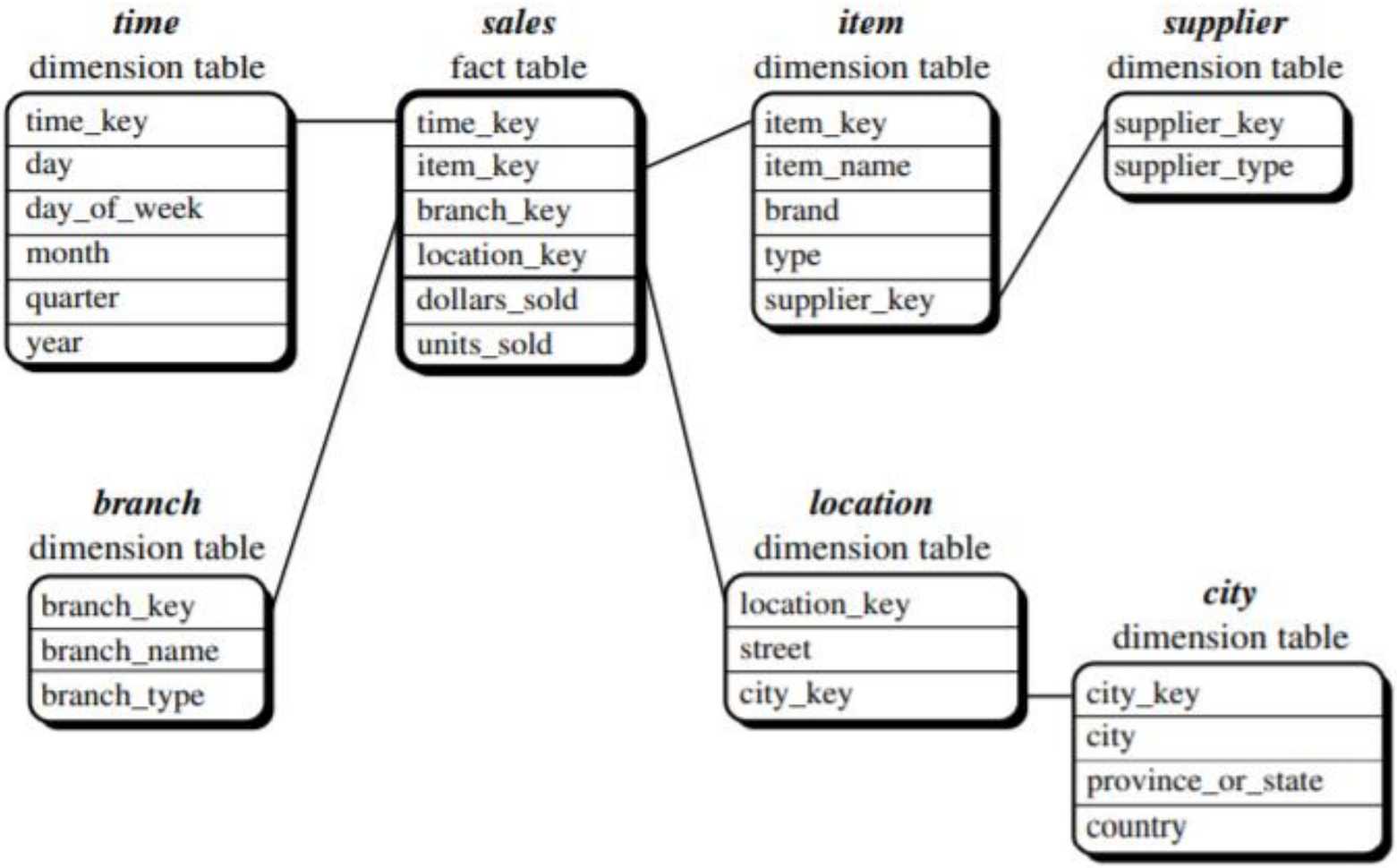
Data Models for Data warehouse

Snowflake Schema

The snowflake schema is a variant of the star schema model, where some dimension tables are normalized, thereby further splitting the data into additional tables

Data Models for Data warehouse

Snowflake Schema



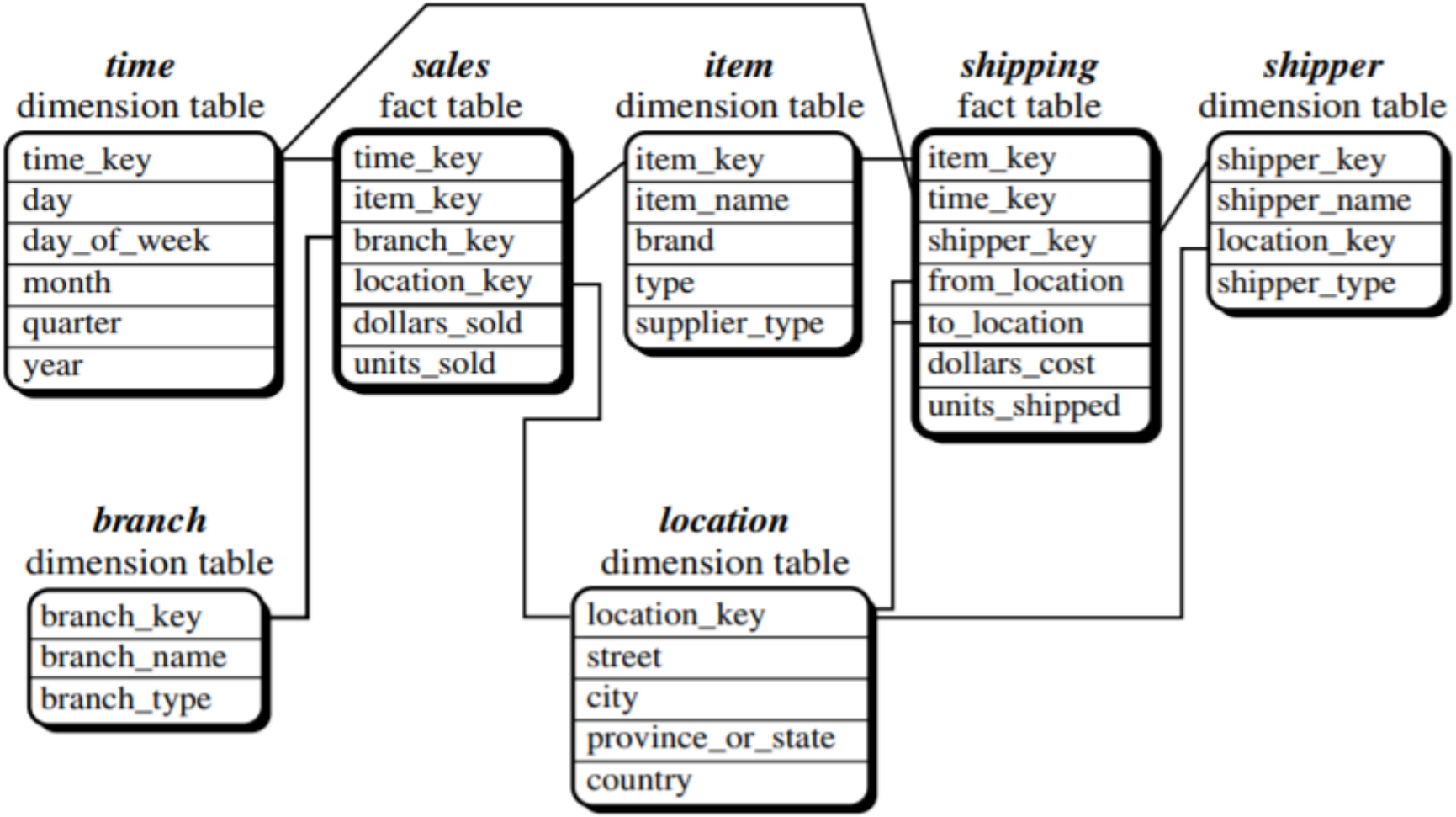
Data Models for Data warehouse

Fact Constellation Schema

"Sophisticated applications may require multiple fact tables to share dimension tables. This kind of schema can be viewed as a collection of stars, and hence is called a galaxy schema or a fact constellation."

Data Models for Data warehouse

Fact Constellation Schema



Data Models for Data warehouse

DMQL for data modelling Schemas:

define cube ⟨cube_name⟩ [⟨dimension_list⟩]: ⟨measure_list⟩

The *dimension definition* statement has the following syntax:

define dimension ⟨dimension_name⟩ as (⟨attribute_or_dimension_list⟩)

Data Models for Data warehouse

Examples of DMQL for data modelling Schemas:

1. Star Schema

```
define cube sales_star [time, item, branch, location]:
        dollars_sold = sum(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)
```

Data Models for Data warehouse

Examples of DMQL for data modelling Schemas:
2. Snowflake Schema

```
define cube sales_snowflake [time, item, branch, location]:
        dollars_sold = sum(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, city, province_or_state, country))
```

Data Models for Data warehouse

3. Fact constellation Schema

```
define cube sales [time, item, branch, location]:
        dollars_sold = sum(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]:
        dollars_cost = sum(cost_in_dollars), units_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
```

Data Models for Data warehouse

Measures (in Fact Tables) : Their Categorization and Computation

1. Distributive Measures- function is applied on each partition
2. Algebraic Measures – algebraic function with M arguments
3. Holistic Measures- an algebraic function is holistic if there is no constant bound on the storage size

Concept Hierarchies

Concept Hierarchies

"A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts"

Ex: Location

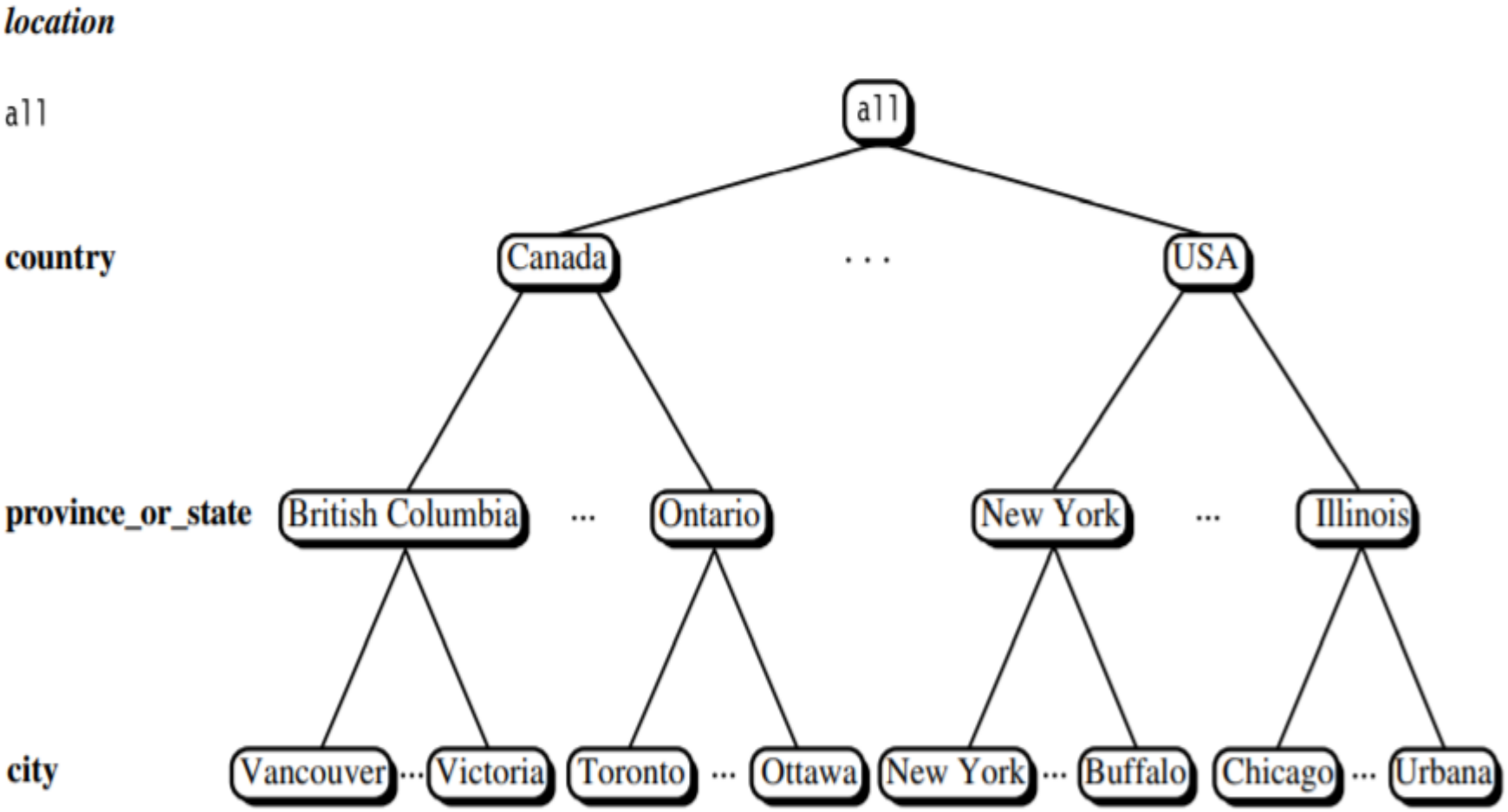
It is defined with Country
It is defined with state, and country
It is defined with city, state, and country
It is defined with street, city, state, and country

Concept Hierarchies

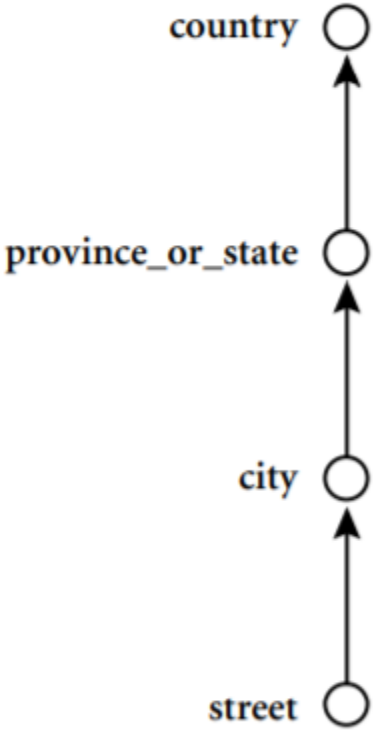
Concept Hierarchies for Location

street < city < province or state < country

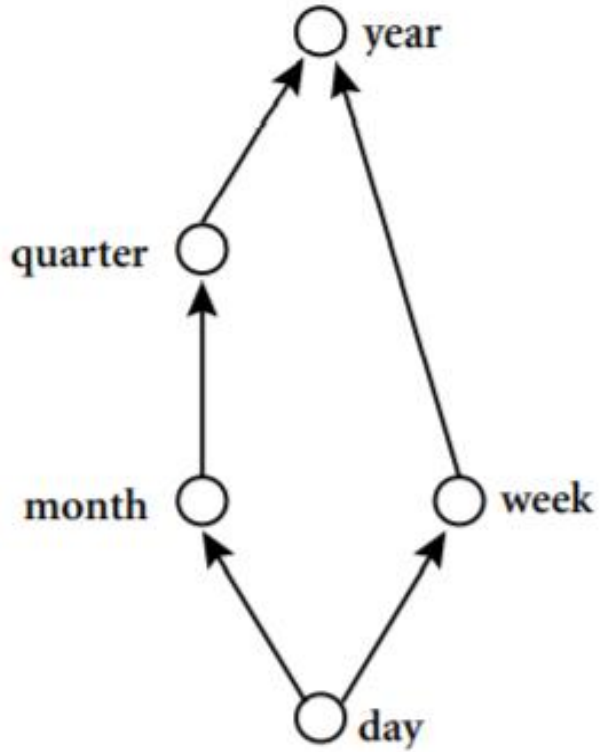
Concept Hierarchies



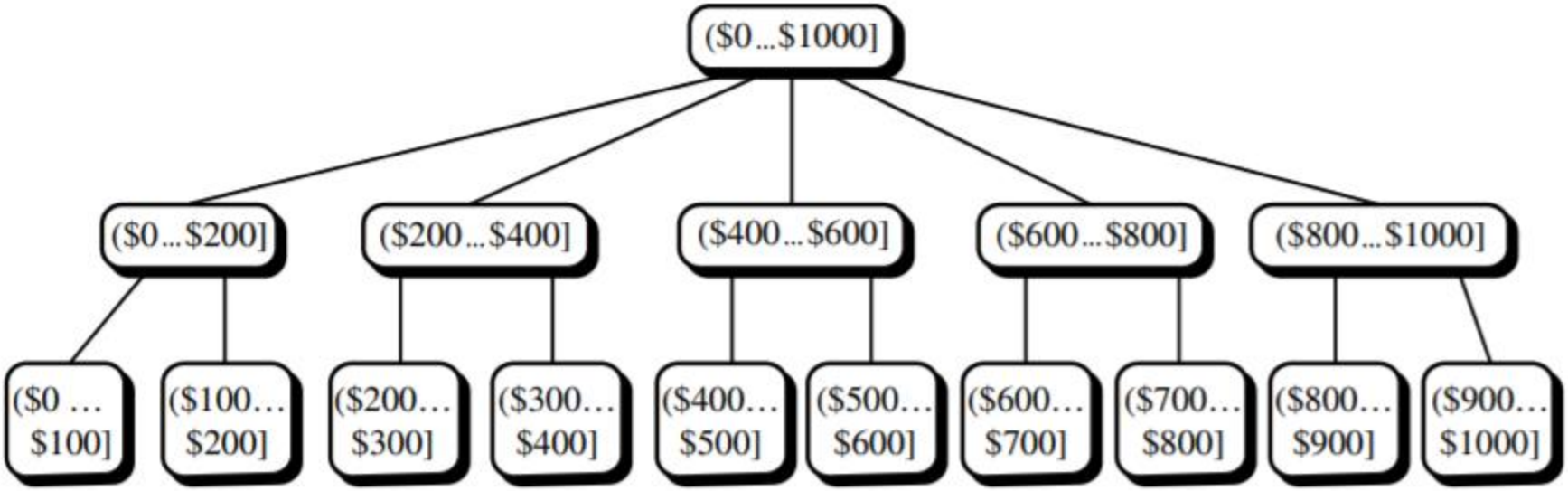
Concept Hierarchies



Concept Hierarchies



Concept Hierarchies



OLAP Operations in the Multidimensional Data Model

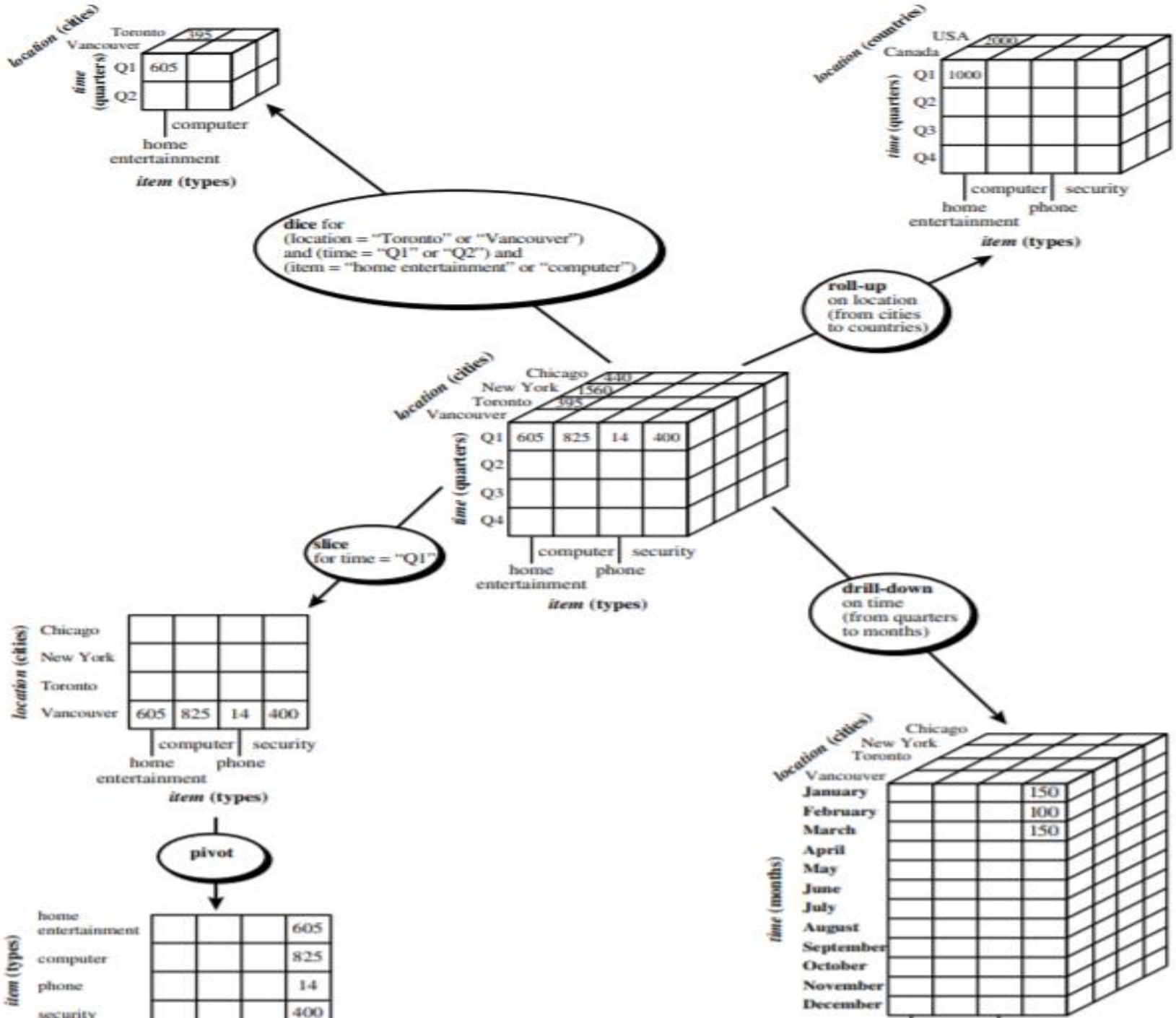
How are concept hierarchies useful in OLAP?

“” In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies”

OLAP Operations in the Multidimensional Data Model

OLAP Operations

1. Roll-Up
2. Drill-down
3. Slice and Dice
4. Pivot


location (cities)
Toronto
Vancouver
395
time (quarters)
Q1
605
Q2
computer
home entertainment
item (types)
location (countries)
USA
Canada
2000
Q1
1000
Q2
Q3
Q4
computer
security
home entertainment
phone
item (types)
dice for
(location = "Toronto" or "Vancouver")
and (time = "Q1" or "Q2") and
(item = "home entertainment" or "computer")
roll-up
on location
(from cities
to countries)
location (cities)
Chicago
440
New York
1560
Toronto
395
Vancouver
Q1
605
825
14
400
Q2
Q3
Q4
time (quarters)
computer
security
home entertainment
phone
item (types)
slice
for time = "Q1"
drill-down
on time
(from quarters
to months)
location (cities)
Chicago
New York
Toronto
Vancouver
605
825
14
400
computer
security
home entertainment
phone
item (types)
pivot
item (types)
home entertainment
605
computer
825
phone
14
security
400
location (cities)
Chicago
New York
Toronto
Vancouver
January
150
February
100
March
150
April
May
June
July
August
September
October
November
December
time (months)