VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

OLAP Database Structures

Lecturer : Dr. Pavle Mogin

SWEN 432

Advanced Database Design and Implementation

Plan for OLAP Database Structures

- Comparing OLAP and OLTP Queries
- Multidimensional data
  - Hypercube and Star Schema
- *Star Schema*
  - *Fact Table*
  - *Dimension Tables*
  - *Attribute Hierarchies*
  - *Constellations*
  - *Readings* :
    - *Ramakrishnan, Gehrke: “Database Management Systems”, Chapter 25, Sections 25.1 and 25.2*
    - *S. Chaudhuri, U. Dayal:*

      An Overview of Datawarehousing and OLAP Technology

OLAP Queries Using OLTP BP

- The contemporary generation of DSS tools is called OLAP
- OLAP transactions are mostly of an ad hoc query type and may involve many thousands of operational database records
- Using operational databases for OLAP would result in unacceptably low performance
  - Since these queries would require joins and computing aggregate functions over large data sets
- OLAP systems need differently structured databases:
  - Multidimensional data and materialized views

OLTP Data

Invoice_Header

InvNo	Date	Customer	Total
2034	15.12.	Vic	3500
2035	15.12.	Massey	1800
2036	16.12.	Vic	2000
2037	16.12.	Massey	800

Invoice_Detail

InvNo	Product	UniPrice	Quantity
2034	mouse	150	20
2034	usb	20	25
2035	mouse	150	12
2036	usb	20	100
2037	usb	20	25
2037	mouse	150	2

OLTP vs OLAP Queries

- OLTP queries are **infrequent** compared to OLAP queries
- An OLTP query:
  - Retrieve the invoice for customer Vic issued on 15 December

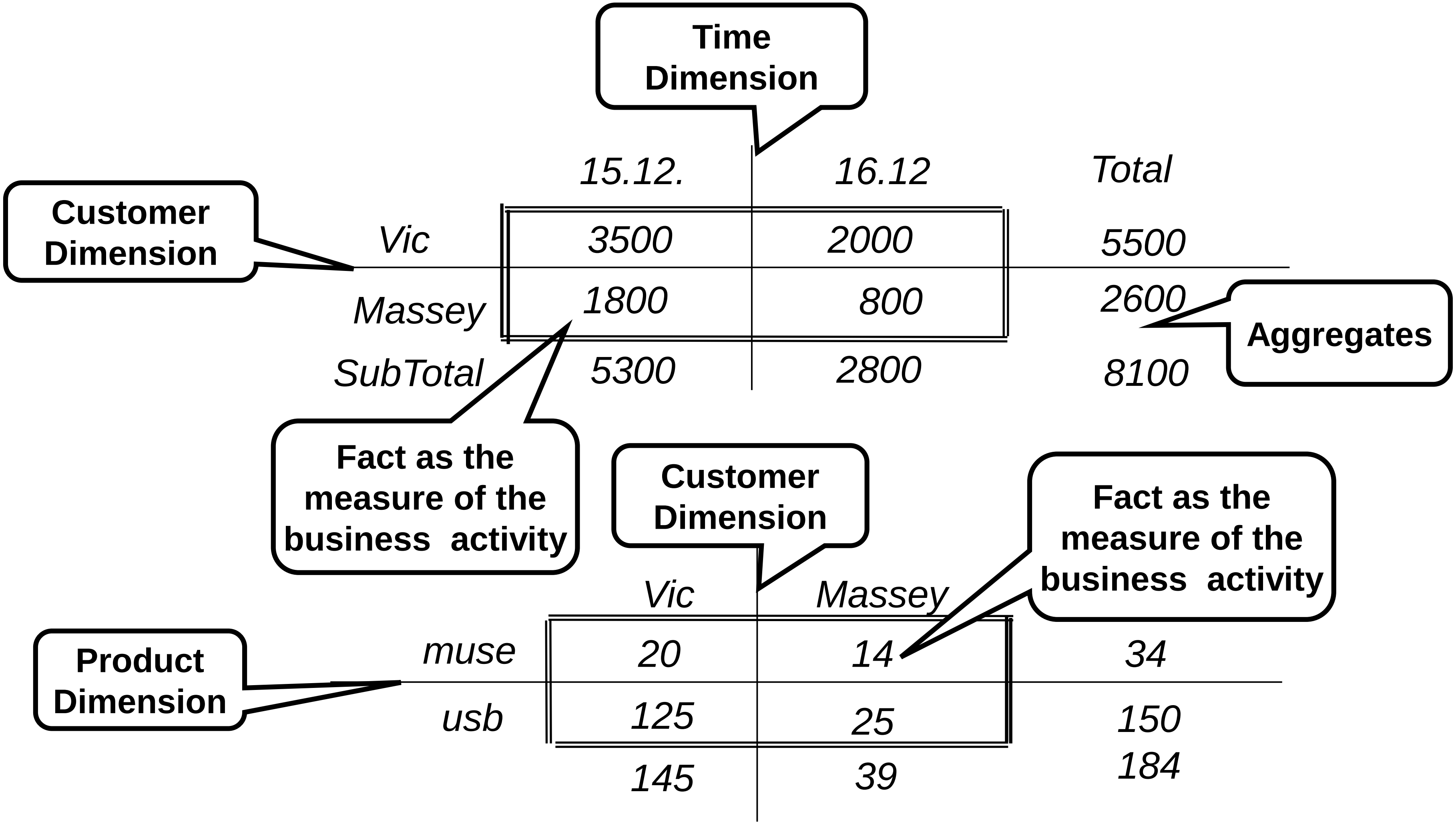
```
select * from invoice_header natural
join invoice_detail where customer =
‘Vic’ and date = ’15.12.’;
```

- OLAP queries:
  - Find the total amount invoiced to Vic in December

```
select sum(unit_price*quantity) from
invoice_header natural join
invoice_detail where customer = ‘Vic’
and substr(date, 4, 2) = ’12’;
```

  - Find the number of mice sold to Vic in December

Multidimensional OLAP Data



Time Dimension

Customer Dimension

	15.12.	16.12	Total
Vic	3500	2000	5500
Massey	1800	800	2600
SubTotal	5300	2800	8100

Aggregates

Fact as the measure of the business activity

Customer Dimension

Fact as the measure of the business activity

Product Dimension

	Vic	Massey	
muse	20	14	34
usb	125	25	150
	145	39	184

OLTP Data Updated

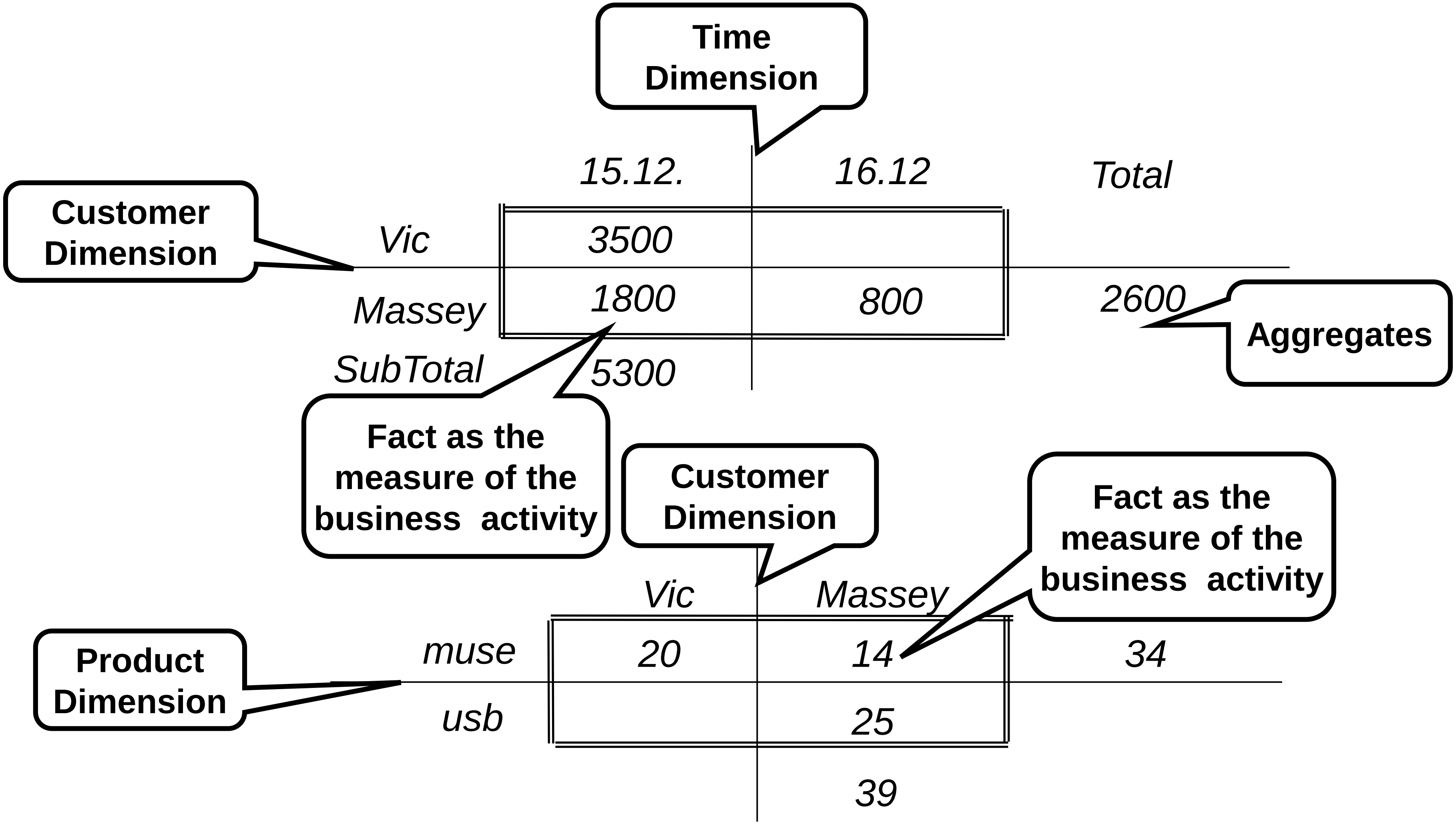
Invoice_Header

InvNo	Date	Customer	Total
2034	15.12.	Vic	3500
2035	15.12.	Massey	1800
2036	16.12.	Vic	2000
2037	16.12.	Massey	800
2038	16.12.	Vic	1800

Invoice_Detail

InvNo	Product	UniPrice	Quantity
2034	mouse	150	20
2034	usb	20	25
2035	mouse	150	12
2036	usb	20	100
2037	usb	20	25
2037	mouse	150	2
2038	usb	20	90

Refreshing Multidimensional Data



Time Dimension

Customer Dimension

	15.12.	16.12	Total
Vic	3500		
Massey	1800	800	2600
SubTotal	5300		

Aggregates

Fact as the measure of the business activity

Customer Dimension

Fact as the measure of the business activity

Product Dimension

	Vic	Massey	
muse	20	14	34
usb		25	
		39	

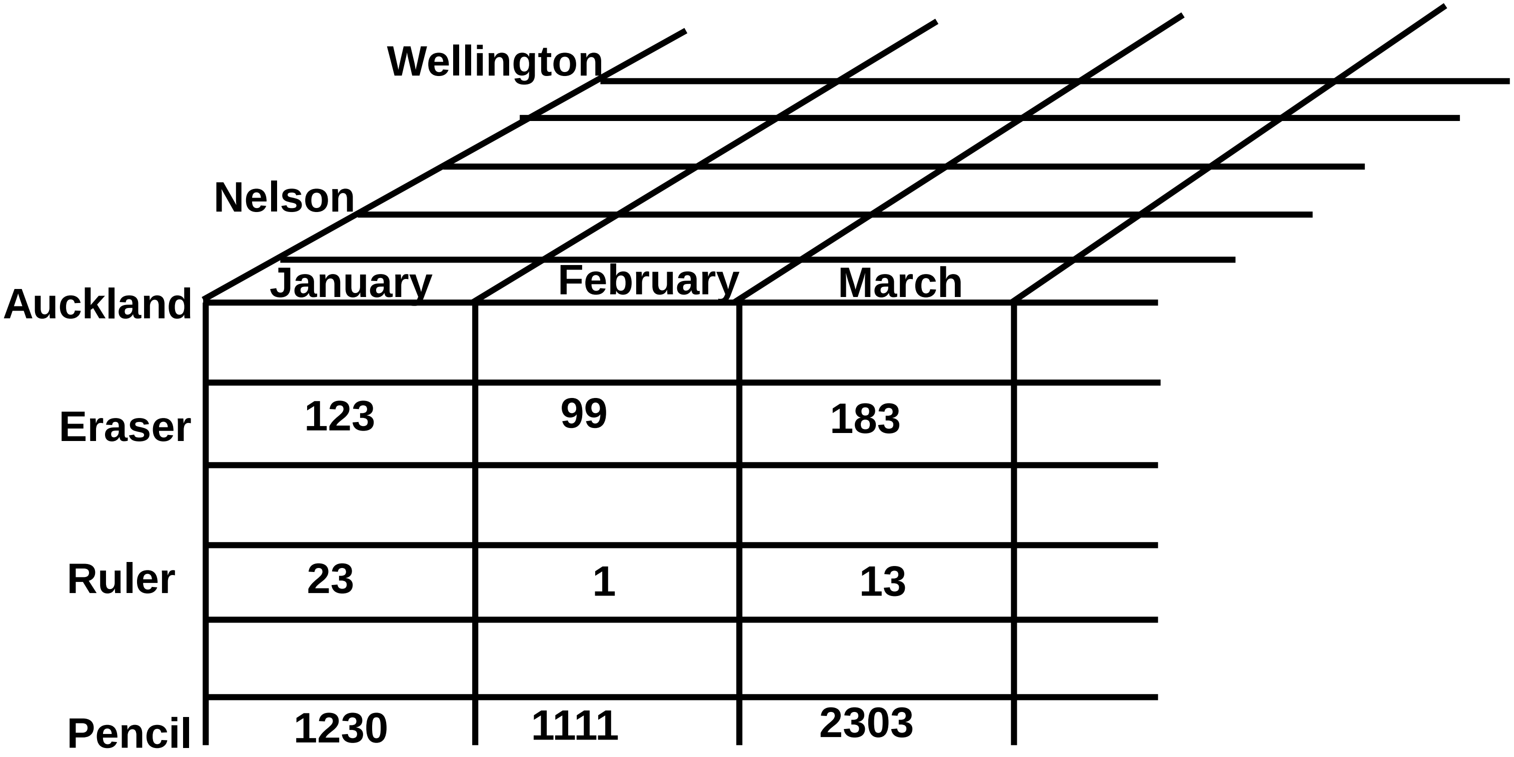
Two Kinds of Multidimensional Structures

- There exist two approaches to building a multidimensional database
- These are:
  - Specialized multidimensional structures, called data cube, and
  - Relational database structures, called star schema

Data Cube

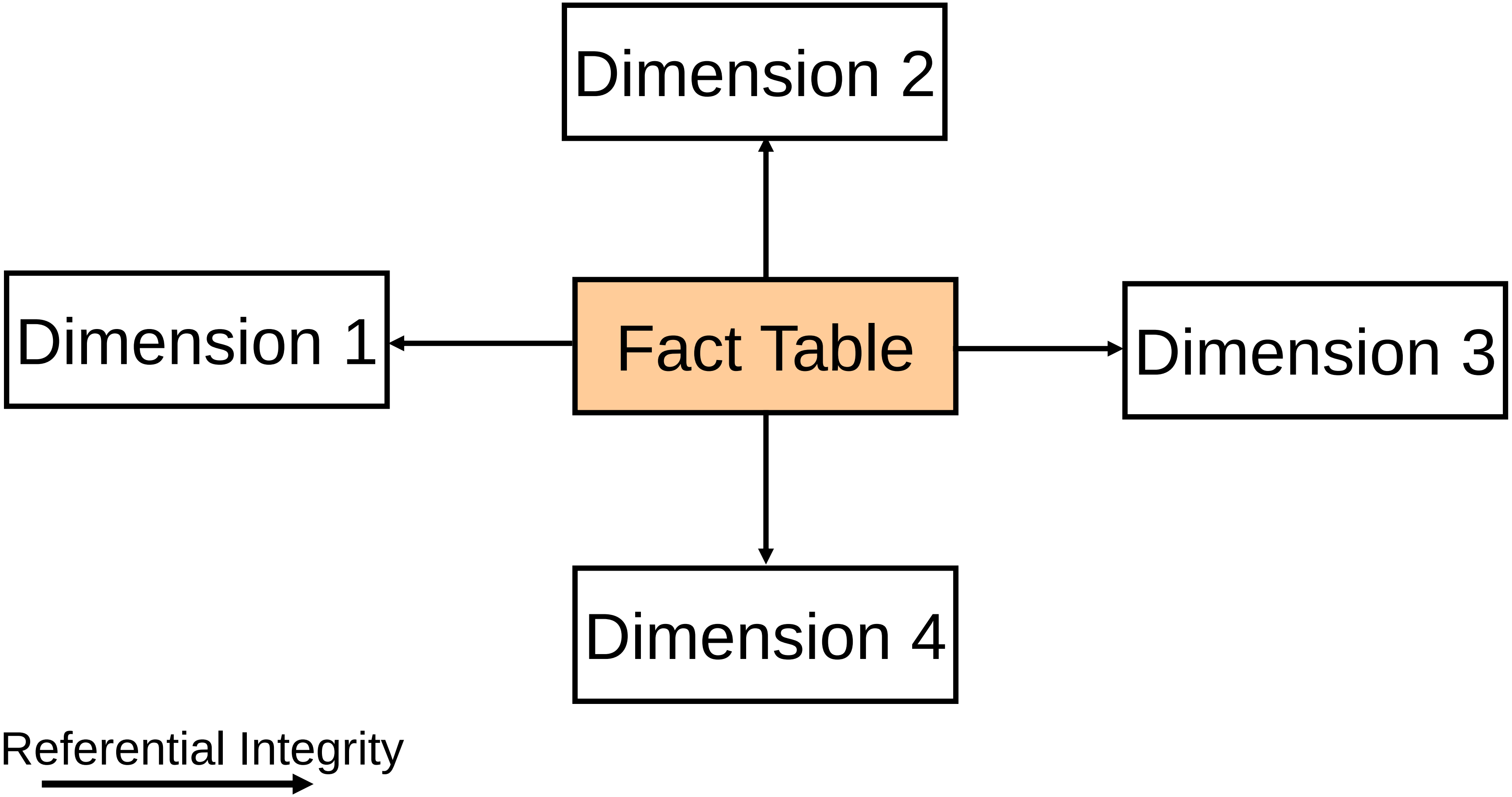
- Data cube is based on so called hypercube data organization
- Every data dimension is associated with one hyper edge of the hypercube
- Facts are recorded in the hyperspace in the points defined by dimension values (that act as coordinates)
- Physically, hypercube is stored in multidimensional arrays
- Hypercube building techniques are non standard and represent vendors’ intellectual property
- By the rule, building a hypercube takes long, but it results in a fast query execution

A Hypercube



	January	February	March
Eraser	123	99	183
Ruler	23	1	13
Pencil	1230	1111	2303

A Star Schema



Relational versus Hypercube

CHARACTERISTICS	*RELATIONAL*	*HYPERCUBE*
DATA ORGANIZATION	*Star Schema* (*standard*)	*Hypercube* (*non standard*)
FLEXIBILITY	*Large* (*unlimited number of dimensions)*	*Low* (*new dimensions require rebuilding of the hypercube*)
DATABASE VOLUME	*Medium to Large*	*Small to Medium*
SPEED	*Good for small,* *Acceptable for medium to large databases*	*Excellent for small and medium,* *Poor for large databases*

Data Mart

- A Data Mart is a small Data Warehouse that satisfies the needs of an organizational unit within a company
- This is a Data Warehouse with only one subject
- Building a Data Mart represents a good pilot project before introducing an integrated Data Warehouse
- Physically, a Data Mart can be built either as a Hyper Cube, or as a Star Schema

Star Schema

- The basic star schema contains four components
- These are:
  - Fact table,
  - Dimension tables,
  - Attributes, and
  - Attribute hierarchies

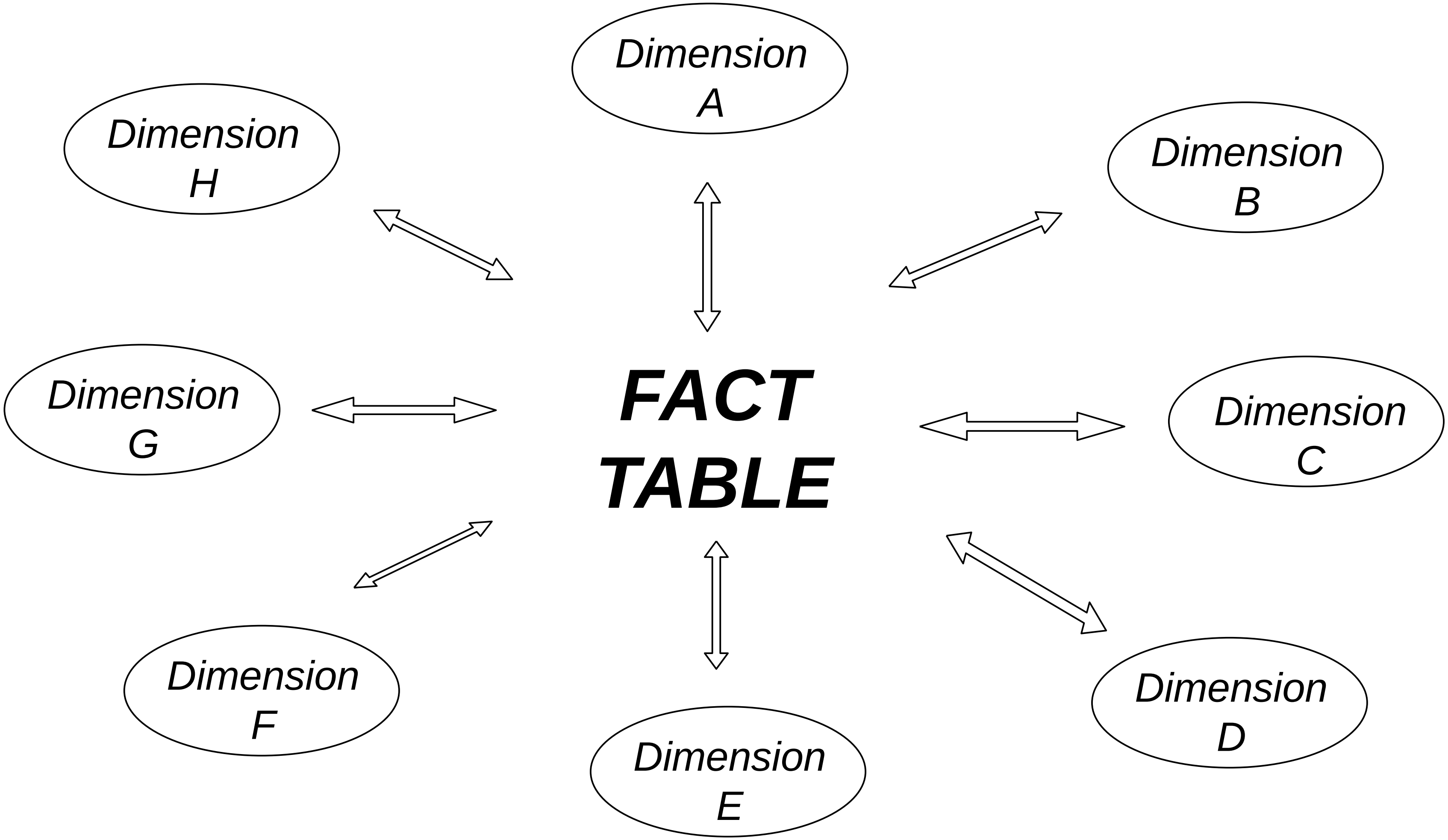
Facts

- Facts are numerical values that represent certain view onto business activities
- Facts represent performance measures of business activities
- Common facts are:
  - Productivity,
  - Expenses,
  - Prices,
  - Sales figures,
  - Profit

Fact Table

- Facts are stored in a ***fact table***
- A fact table is also called ***detail table***
- Every fact table constitutes a mid point of a star schema
- Fact table is periodically updated by inserting aggregated data from operational databases
- Facts can also be calculated in the course of a query execution
- Every fact table is associated with corresponding dimension tables

Star Schema – a Principle View



Dimensions

- Dimensions are characteristics of facts
- They describe the context of facts
- Facts are associated with dimensions
- Example:
  - If sales of products in locations during a time period is a fact, then its dimensions are:
    - Products,
    - Locations, and
    - Time
- Dimensions are stored in dimension tables

Attributes

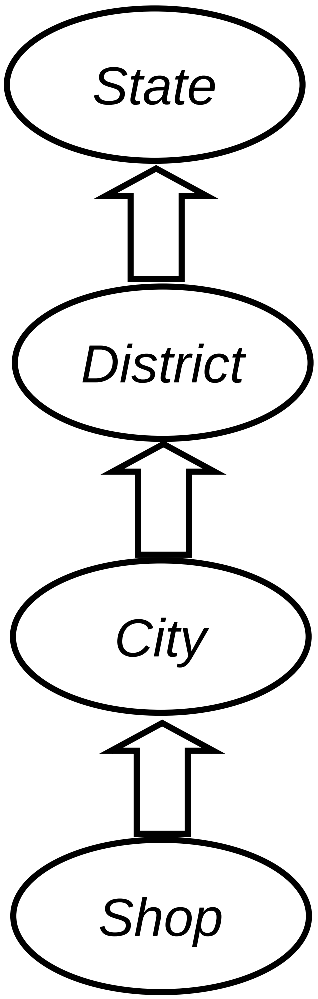
- Attributes are properties of dimensions
- They are used to search, retrieve, and classify facts
- Dimensions contain only those attributes that are used in the decision making process
- Example:
  - If product, location, and time are sales dimensions, then the possible attributes may be:
    - For the product: *ProductId*, *ProdName*, *Prod_Type*, *Supplier,*
    - For the location: *District*, *City*, *ShopId*
    - For the time: *Year*, *Quarter*, *Month*, *Week*, *DayId*

Attribute Hierarchy

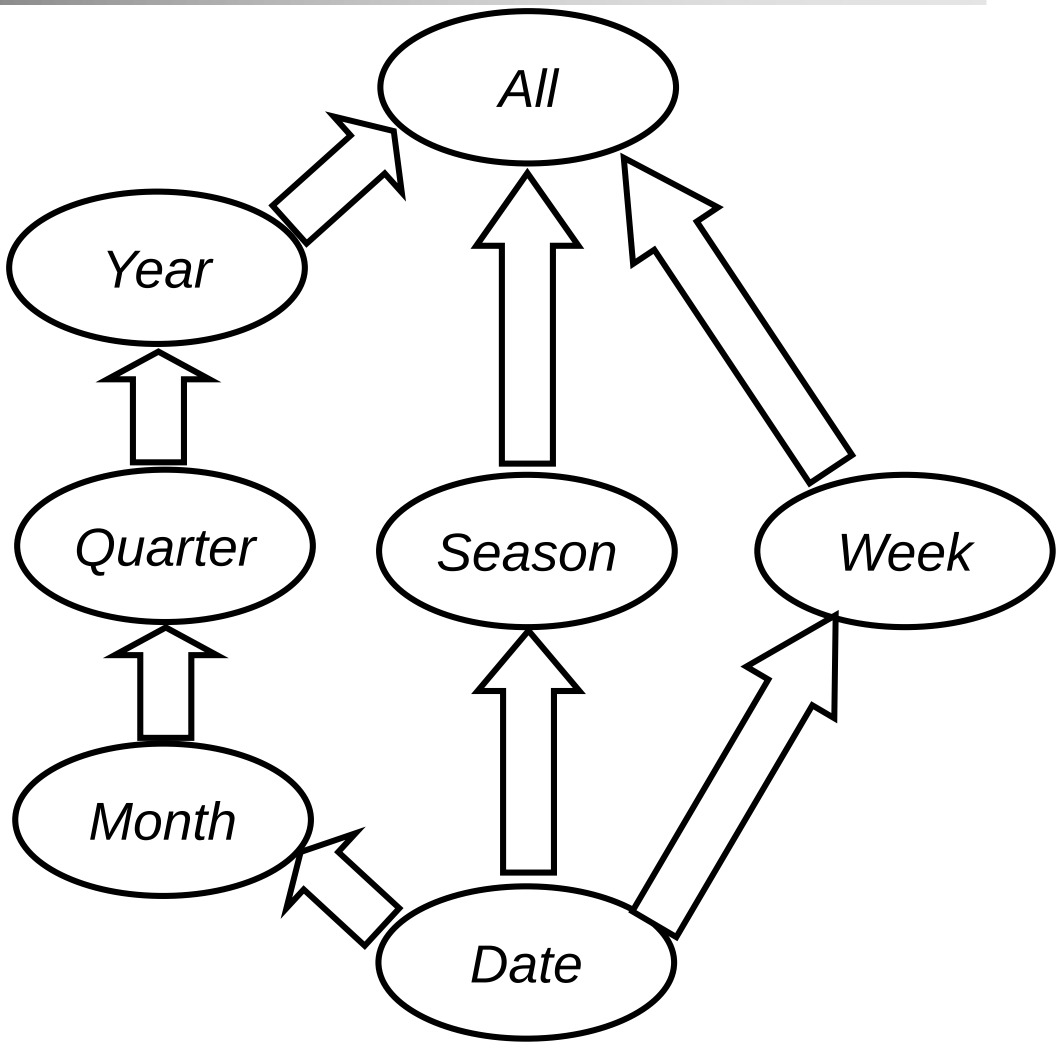
- Attributes can be arranged in a hierarchic structure
- The relationship between hierarchy levels is N:1
- An attribute hierarchy determines a sequence of functional dependencies
- For example, a product hierarchy:
  - *Product→Product_Type*
  - *Product_Type →Industry*

Attribute Hierarchy (some examples)

Location hierarchy



Time hierarchy



Use of Attribute Hierarchies

- Attribute hierarchies are used:
  - To analyze facts at various aggregation levels, usually starting from a higher one, and
  - Also, to enhance query rewriting
- Here, we focus on analysis:
  - If an analysis shows significant differences in the yearly sales in an industry, we can use the corresponding attribute hierarchy to find products that mainly contributed to the difference

Drill Down Example

Industries	2000	2001	2002
Auto Industry	1500	940	1480
Food Industry	3500	3750	3650
Wood Industry	900	1000	950

Auto Industry	2000	2001	2002
Truck	300	250	330
4wd	500	450	500
Limousine	700	240	650

Limousines	2000	2001	2002
Escort	500	40	370
Falcon	100	100	130
Mustang	100	100	150

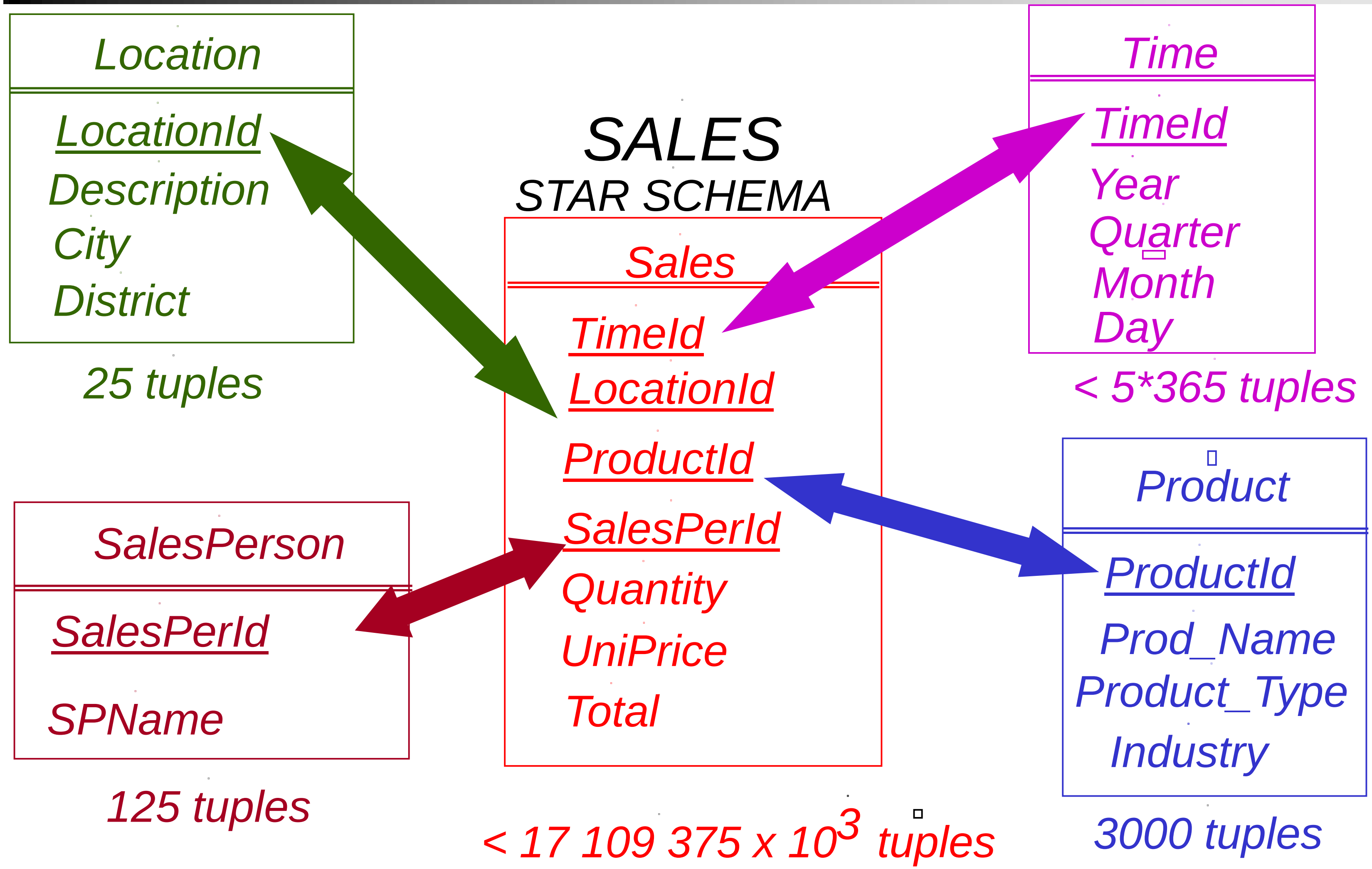
Data Aggregation Ways

- An attribute hierarchy defines the ways of data aggregation (getting data of coarser granularity), or how to look for more precise individual data (getting data of finer granularity)
- The corresponding operations are called roll-up and drill down, respectively
- Not all columns of a dimension table are included in an attribute hierarchy
- Some are used just for mere dimension description

Representing a Star Schema

- Facts and dimensions are stored in relational tables
- A tuple of a fact table contains at least one fact value and primary key values of each of its dimension tables
- Fact values are also called ***measures***
- Primary key values of dimension tables define the context of a fact value
- A fact table primary key is the union of its dimension primary keys
- A fact table is often in BCNF, but dimension tables are usually not
- Dimension tables are also called look-up or reference tables

Star Schema – an Example



SALES

STAR SCHEMA

Location

LocationId
Description
City
District

25 tuples

Time

TimeId
Year
Quarter
Month
Day

< 5*365 tuples

Sales

TimeId
LocationId
ProductId
SalesPerId
Quantity
UniPrice
Total

< 17 109 375 x 10^3 tuples

SalesPerson

SalesPerId

SPName

125 tuples

Product

ProductId
Prod_Name
Product_Type
Industry

3000 tuples

Extension of a Star Schema

Location

LocId	City
1	Well
2	Nels
3	Auck

SalesPerson

PerId	Name
1	John
2	Susan
3	James
4	Susan
5	Ann

Sales

LocId	PerId	TimId	Amnt
1	1	1	230
1	1	2	300
1	1	8	310
1	2	7	50
2	3	1	550
2	3	5	100
3	4	6	880
3	5	1	60
3	5	2	60
3	5	4	140

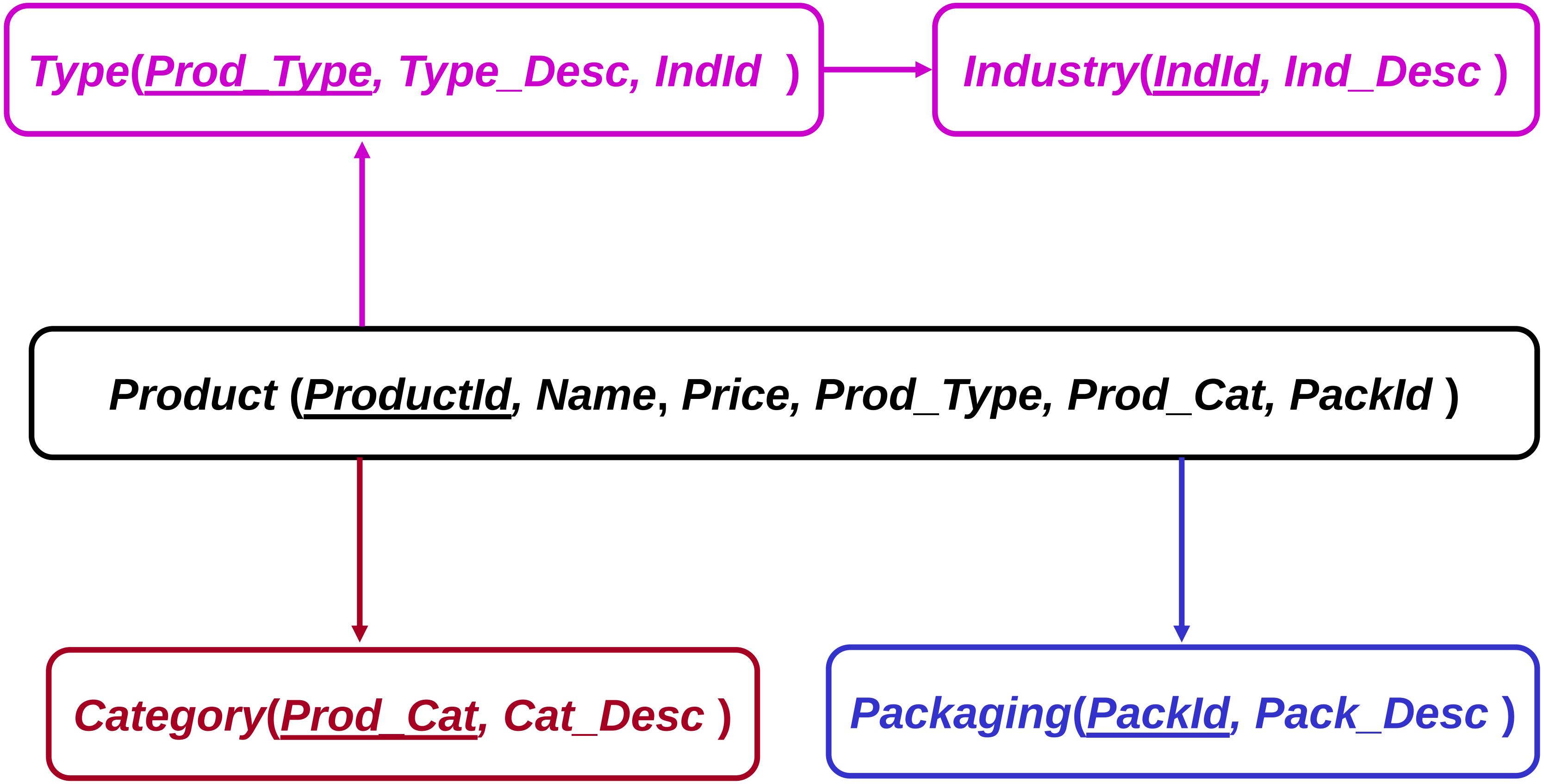
Time

TimId	Day
1	Mon
2	Tue
3	Wed
4	Thu
5	Fri
6	Sat
7	San
8	Mon

Snowflake Schema

- A snowflake schema is a variation of the star schema in which dimension tables are in the third or in BCNF normal form
- By the normalization, each dimension attribute hierarchy is split into a number of relation schemas
- These relation schemas are associated by primary key / foreign key pairs

A Snowflake Structure



Type(Prod_Type, Type_Desc, IndId)

Industry(IndId, Ind_Desc)

Product (ProductId, Name, Price, Prod_Type, Prod_Cat, PackId)

Category(Prod_Cat, Cat_Desc)

Packaging(PackId, Pack_Desc)

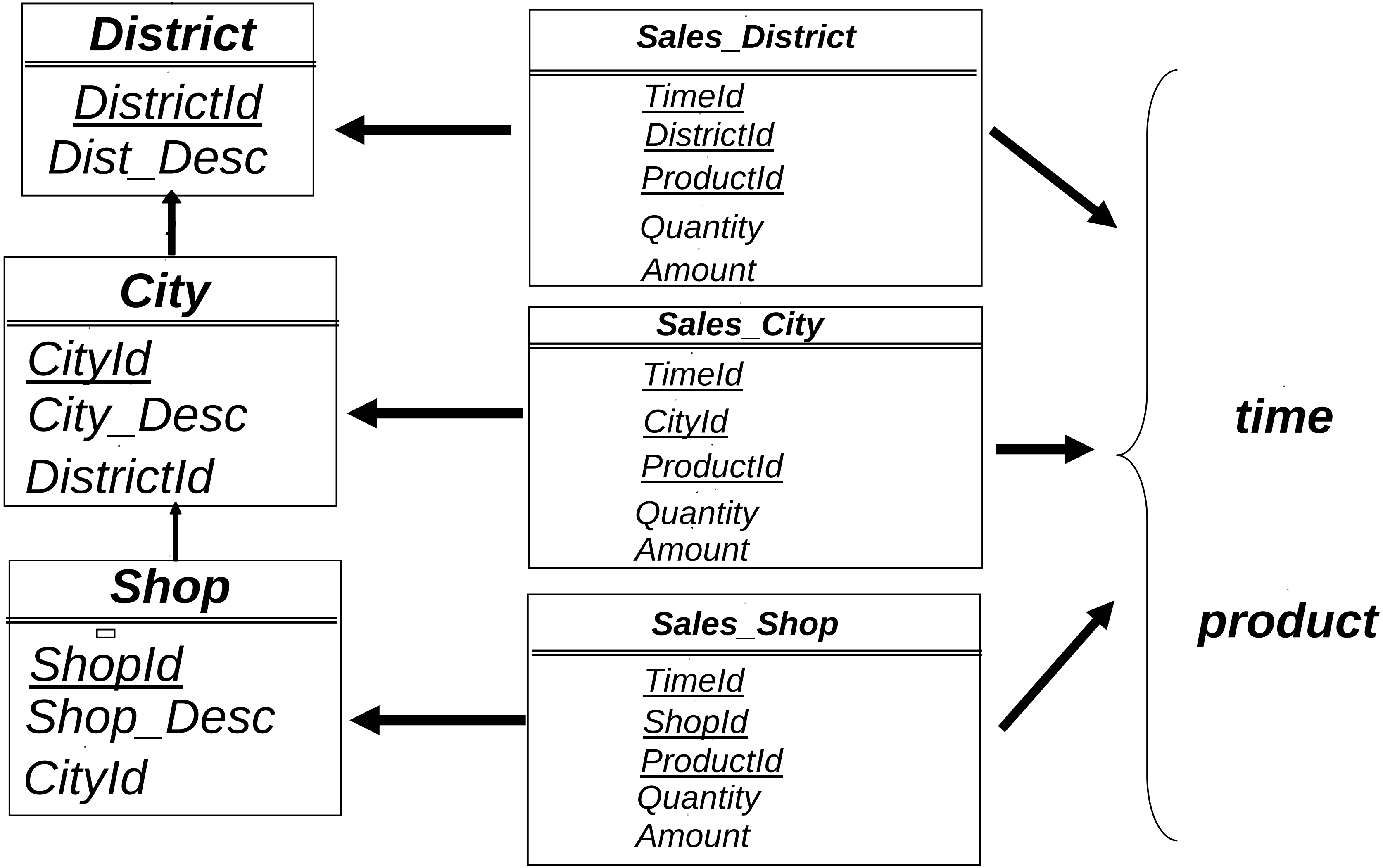
Star Constellation Schema

- By the rule, a Data Warehouse contains a larger number of fact tables
- Every fact table is dedicated to a range of specific decision making tasks
- The same dimension table can represent a component of more than one star schema
- Example:
  - Suppose sales and orders constitute two Data Warehouse subjects, represented by two star schemas
  - Time, products, and customers represent dimensions of the both star schemas, and they would share the same instances of these dimensions

Multiple Aggregate Level Schemas

- The increase of the number of star schemas in a Data Warehouse can also be a consequence of the need to enhance the performance by increasing the number of data aggregation levels
  - In this case, there is more star structures with different aggregation levels within the same subject generated during Data Warehouse loading and refreshing
- To allow this kind of a constellation, at least one dimension has to be snow-flaked

A Multiple Aggregates Constellation



District

DistrictId
Dist_Desc

City

CityId
City_Desc
DistrictId

Shop

ShopId
Shop_Desc
CityId

Sales_District

TimeId
DistrictId
ProductId
Quantity
Amount

Sales_City

TimeId
CityId
ProductId
Quantity
Amount

Sales_Shop

TimeId
ShopId
ProductId
Quantity
Amount

time

product

Summary (1)

- On – Line Transaction Processing (OLTP):
  - Day – to – day clerical tasks
  - Atomic transactions that mainly update operational normalized databases
- On – Line Analytical Processing (OLAP):
  - Decision making
  - Ad – hoc queries, that require many joins and aggregates if a OLTP database has been used
- Hypercube is highly specialized proprietary multidimensional data structure
- Star schema is relational multidimensional data structure
- Data Mart is a small DW with just one subject

Summary (2)

- A fact table contains measures and dimension keys
- A fact table is usually BCNF
- Dimension tables contain dimension attributes and are usually not BCNF
- A star schema contains at least one fact table and a number of dimension tables
- Attribute hierarchies are used to make aggregates
- A snowflake schema contains BCNF dimension tables, where dimension attribute hierarchies are used to decompose dimensions
- A constellation schema contains many stars