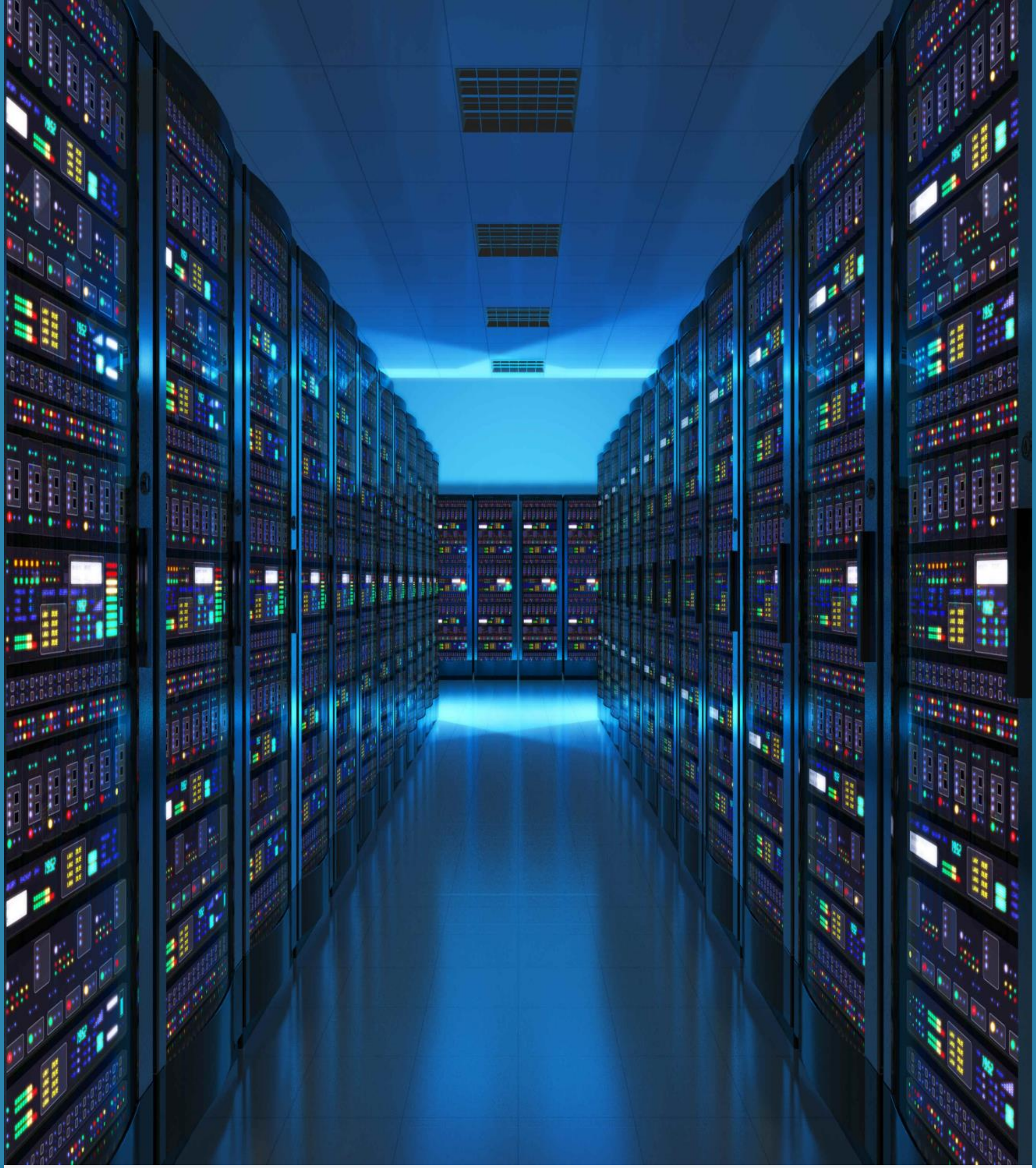


KHO VÀ KHAI PHÁ DỮ LIỆU

(DATA WAREHOUSE AND DATA MINING)

GV: Phan Đình Vấn

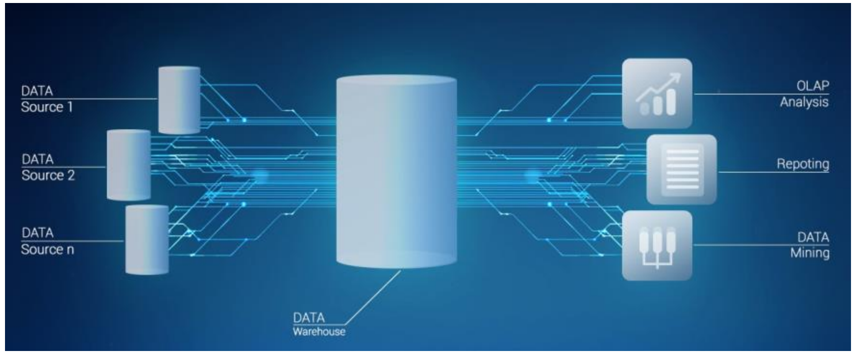
DATA WAREHOUSE

- Định nghĩa kho dữ liệu
- Mục đích và ý nghĩa của kho dữ liệu
- Đặc tính của dw
- Demo ETL

1.1. ĐỊNH NGHĨA KHO DỮ LIỆU (DATA WAREHOUSE - DW)

- Đầu 1990s, Bill Inmon đã đặt ra thuật ngữ kho dữ liệu:
  - A data warehouse is a collection of
    - Subject-oriented,
    - Integrated,
    - Time-variant,
    - Non-volatile

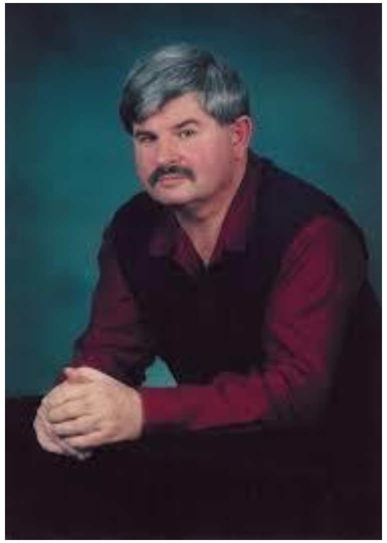
    data in support of management decisions.



- Ralph Kimball định nghĩa kho dữ liệu:
  - A DW is a copy of transaction data specifically structured for query and analysis.

1.1. ĐỊNH NGHĨA DW

- **Bill Inmon**, the Father of Data Warehousing
  - William H. Inmon (born 1945) is an American computer scientist
  - Bill Inmon first began to discuss the principles around the Data Warehouse.
  - In 1992, Inmon published **Building the Data Warehouse**.
  - In 2007, Inmon was named by Computerworld as one of the "Ten IT People Who Mattered in the Last 40 Years."
  - Inmon's approach to Data Warehouse design focuses on a centralized data repository modeled to the third normal form. (Top-Down)



1.1. ĐỊNH NGHĨA DW

- **Ralph Kimball**, other Father of Data Warehousing
  - 1996, Ralph Kimball published **The Data Warehouse Toolkit**
  - Kimball's early career in IT in the 1970s was highlighted by work as a key designer for the Xerox Star Workstation
  - In the 1980s, he work with decision support systems as a Vice President for Metaphor Computer Systems.
  - 1986, founded Red Brick Systems company with a full-fledged Data Warehouse application served as a major product.
  - 1992, left Red Brick and start his own consultancy, Ralph Kimball Associates
  - His well-regarded series of Data Warehouse Toolkit books soon followed.
    - Web-based Data Warehousing
    - ETL in a Data Warehousing environment,
    - Microsoft-specific editions that cover SQL Server and the Microsoft Business Intelligence Toolset.

1.1. ĐỊNH NGHĨA DW

- DW là tập các phương pháp, kỹ thuật và công cụ có thể kết hợp lại để cung cấp thông tin cho người dùng dựa trên việc tích hợp dữ liệu từ nhiều nguồn, nhiều môi trường khác nhau. (John Ladley)

- Data Warehouse Technology is a set of methods, techniques and tools that can combine and support each other to provide information to the user based on integration data from different multiple sources/Environments. (John Ladley)

1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU

- Sự bùng nổ của dữ liệu
- Sự phức tạp của dữ liệu
- Sự phân cấp và phân tán của hệ thống
- Vấn đề quản lý dữ liệu lịch sử
- Vấn đề lập báo cáo
- Vấn đề về quản lý và chia sẻ dữ liệu
- Hỗ trợ ra quyết định

1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU

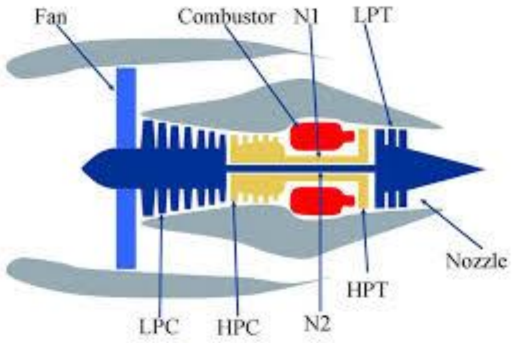
- Kho dữ liệu là một cơ sở dữ liệu được thiết kế để hỗ trợ các hoạt động kinh doanh thông minh (BI)
  - Giúp người dùng hiểu và nâng cao hiệu quả của tổ chức của họ.
  - Nó được thiết kế để truy vấn và phân tích hơn là để xử lý giao dịch
  - Thường chứa dữ liệu lịch sử được lấy từ dữ liệu giao dịch
  - Kho dữ liệu tách biệt việc phân tích và các công việc giao dịch
  - Cho phép hợp nhất dữ liệu từ nhiều nguồn.

1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU

- DW cũng là cơ sở dữ liệu quan hệ, môi trường kho dữ liệu có thể bao gồm:
  - Extraction, transportation, transformation, and loading (ETL)
  - Các công cụ phân tích thống kê, báo cáo, khai thác dữ liệu
  - Các ứng dụng khác quản lý quá trình thu thập dữ liệu, chuyển thành thông tin hữu ích, cung cấp nó cho người dùng.

1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU

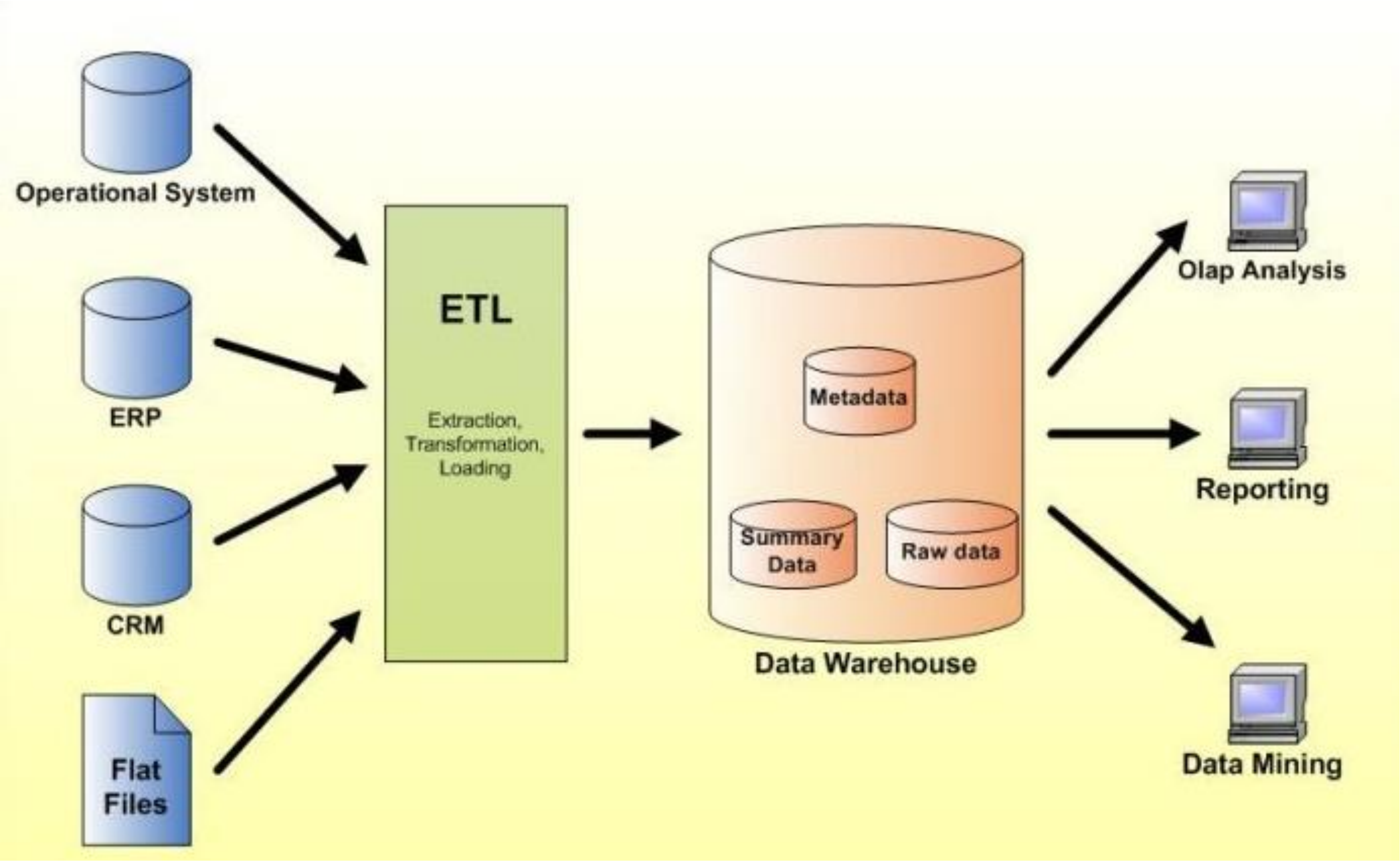
- Để đạt được mục tiêu nâng cao kinh doanh thông minh (trí tuệ kinh doanh), kho dữ liệu cần được thu thập từ nhiều nguồn.
  - Từ nội bộ
  - Từ các phần mềm (đã mua)
  - Từ bên thứ ba của các công ty cung cấp dữ liệu
  - Các nguồn khác.
  - Dữ liệu có thể liên quan đến giao dịch, sản xuất, tiếp thị, nguồn nhân lực…
  - Ngày nay, dữ liệu có thể là từ các mạng xã hội, web... (click, like…)
  - Dữ liệu từ các cảm biến (sensor) được tích hợp trong máy móc phức tạp.



1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU

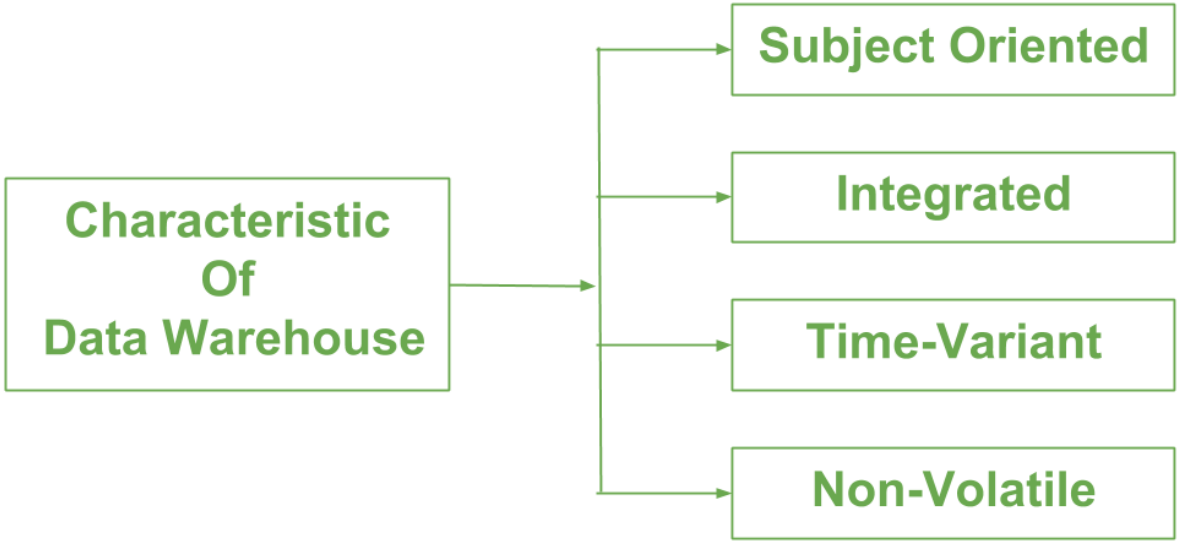
- Kho dữ liệu khác biệt với hệ thống xử lý giao dịch trực tuyến (OLTP).
  - Kho dữ liệu: tách biệt công việc phân tích với công việc giao dịch.
  - DW định hướng đọc hơn là ghi và chỉnh sửa.
    - Tăng hiệu suất phân tích
    - Tránh ảnh hưởng đến hệ thống giao dịch.
  - DW có thể được tối ưu hóa để hợp nhất dữ liệu từ nhiều nguồn và trở thành nguồn duy nhất của tổ chức.
  - Nguồn dữ liệu nhất quán cho tất cả người dùng
  - Ngăn ngừa được các tương tranh dữ liệu, nâng cao hiệu quả khi ra quyết định.

1.2. MỤC ĐÍCH VÀ Ý NGHĨA CỦA KHO DỮ LIỆU



1.3. ĐẶC TÍNH CỦA DW

- Hướng chủ đề (Subject-Oriented)
- Tích hợp (Integrated)
- Dữ liệu gắn thời gian và có tính lịch sử (Time-Variant)
- Dữ liệu không biến động (Non-volatile)



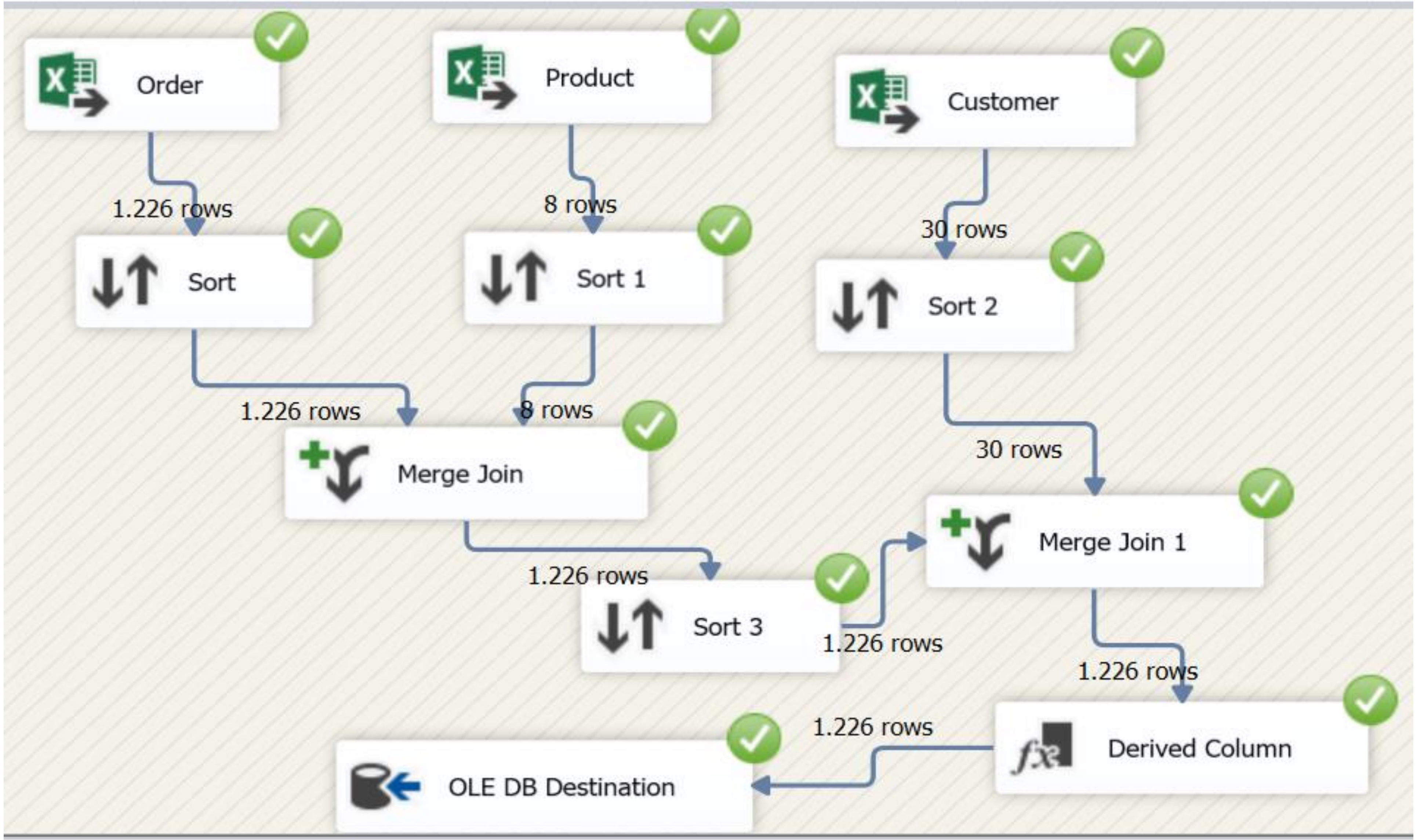
1.3. ĐẶC TÍNH CỦA DW

- Hướng chủ đề (Subject-Oriented)
  - Dữ liệu trong DW được xác định từ đầu là để phân tích về một hoặc một số chủ đề nhất định, ví dụ “doanh thu”.
  - DW không phải là nơi lưu trữ thông tin về mọi mặt hoạt động của công ty hoặc tổ chức.
- Tích hợp (Integrated)
  - Dữ liệu được tập hợp từ nhiều nguồn và lưu trữ nhất quán. Ví dụ, cùng một mặt hàng được quản lý với 2 tên khác nhau ở 2 hệ thống quản lý kho và hệ thống bán hàng.
  - Khi tập hợp dữ liệu về DW, sẽ có một bước biến đổi (transform) để đưa về một tên duy nhất cho mặt hàng này.

1.3. ĐẶC TÍNH CỦA DW

- Biến đổi theo thời gian (Time-variant)
  - Dữ liệu trong Data Warehouse luôn gắn với một thời điểm cụ thể trong một giới hạn thời gian nhất định. Ví dụ, người sử dụng dữ liệu có thể truy vấn lịch sử hàng tồn kho trong 3, 6 hoặc 12 tháng trước.
  - Đây cũng là một trong những khác biệt căn bản của DW với các hệ thống OLTP, vốn chỉ lưu trữ trạng thái dữ liệu mới nhất (trong ví dụ trên là lượng hàng tồn kho hiện tại).
- Ổn định (Non-volatile)
  - Một khi đã được đưa vào Data Warehouse, dữ liệu sẽ không bị thay đổi hoặc xóa.
  - Đặc điểm này cho phép người quản lý có được bức tranh toàn cảnh về toàn bộ lịch sử hoạt động.

DEMO ETL

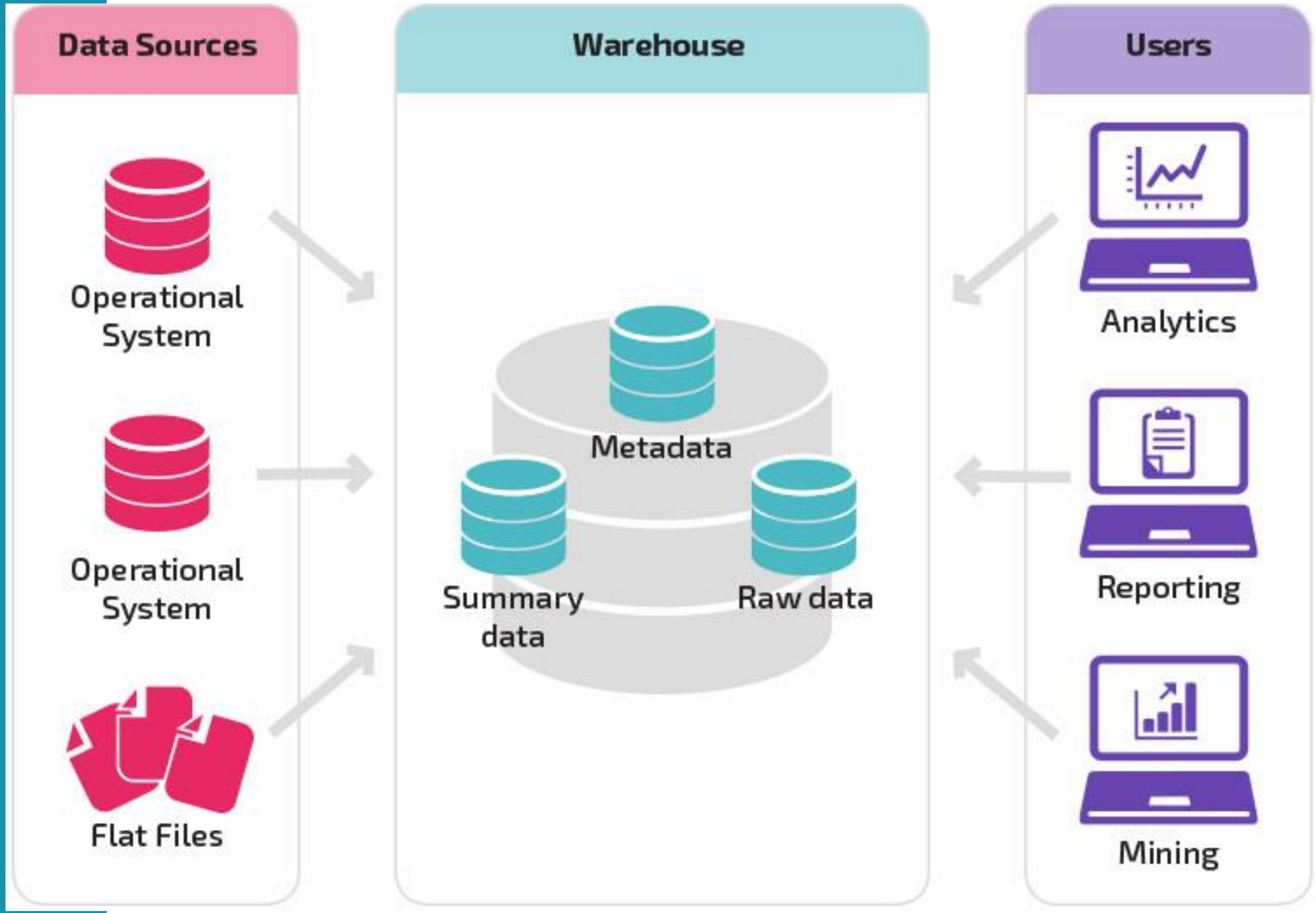

Order
Product
Customer
1.226 rows
8 rows
30 rows
Sort
Sort 1
Sort 2
1.226 rows
8 rows
Merge Join
30 rows
Merge Join 1
1.226 rows
Sort 3
1.226 rows
1.226 rows
1.226 rows
Derived Column
OLE DB Destination

Questions

1. What type of data are there in the world?
2. What sources can create data?
3. How can connect, process and analyze these types of data effectively?
4. How can we store these types of data effectively?
5. What solutions and tools to support for storing and analyzing those data?
6. What is a data warehouse?
7. What are characteristics of a data warehouse?
8. What are the purposes of data warehouse?
9. Who is the father of the data warehouse?
10. How is the history of the data warehouse?

DATA WAREHOUSE

- Database vs. Data warehouse
- Operation database vs. Data warehouse
- Data warehouse architectures
- Data warehouse building process



1.4. DATABASE VS. DATA WAREHOUSE

Database	Data Warehouse
Transaction Oriented	Subject Oriented
Detail Data	Summarized Data
For OLTP	For OLAP
Optimized for write operation	Optimized for read operation
Low performance for analytical queries	High performance for analytical queries
Current/ Real - time	Historical
Size data: MB - GB	Size data: GB-TB
Purpose for data retrieval, Updating and management	Purpose for data analysis and decision making

1.4. DATABASE VS. DATA WAREHOUSE

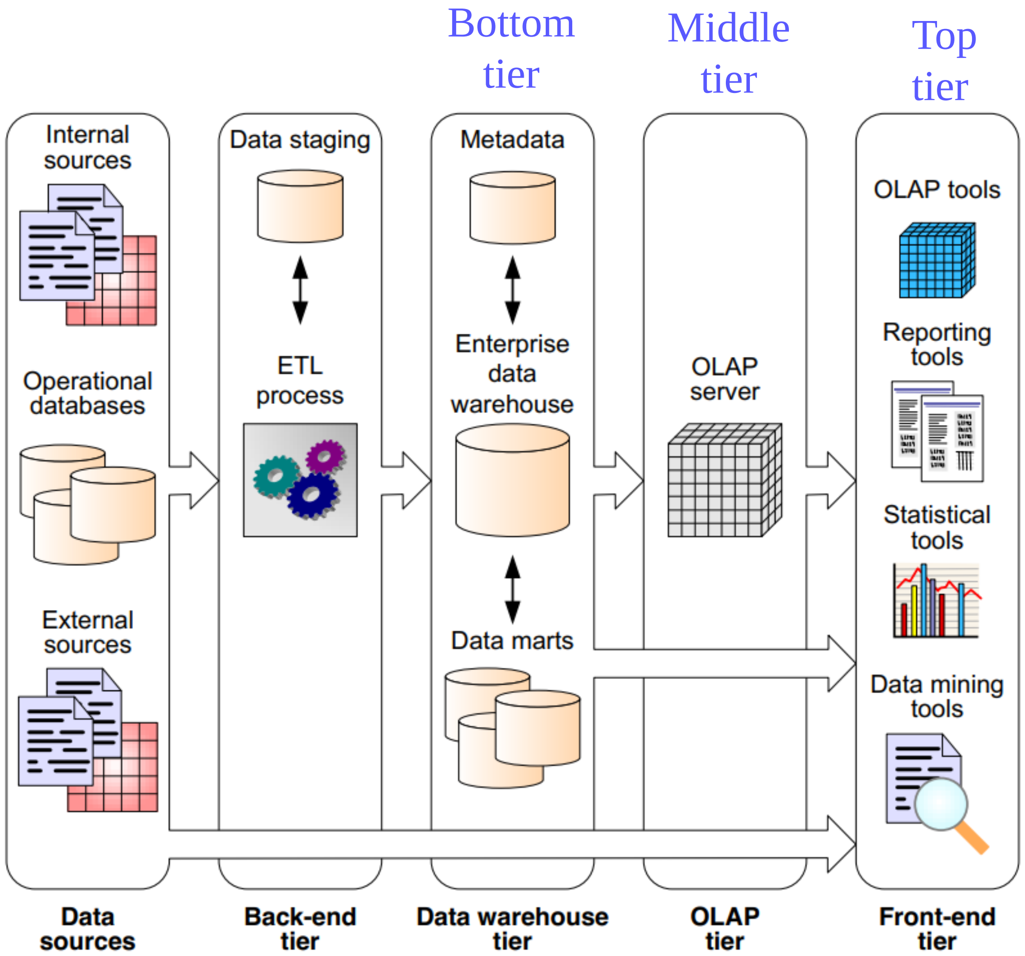
Database	Data Warehouse
Is designed to record	Is designed to analyze
Tables and joins of a database are complex as they are normalized.	Table and joins are simple in a data warehouse because they are denormalized.
Generally limited to a single application	Stores data from any number of applications
ER model	Star schema, snowflake
Capture data	Analyze data

1.4. OPERATION DATABASE VS. DATA WAREHOUSE

- **Operation database**
  - Cơ sở dữ liệu tác nghiệp là nguồn cho kho dữ liệu.
  - Bao gồm thông tin chi tiết được sử dụng để điều hành hoạt động hàng ngày của doanh nghiệp.
  - Dữ liệu thường xuyên thay đổi.
  - Quản lý dữ liệu động trong thời gian thực
  - Cơ sở dữ liệu hoạt động được gọi là OLTP (xử lý giao dịch trực tuyến).
- **Data warehouse**
  - DW mục đích để phân tích dữ liệu và ra quyết định, còn được gọi là hệ thống Xử lý phân tích trực tuyến (OLAP).
  - OLAP và OLTP đều là cơ sở dữ liệu quan hệ, nhưng mục tiêu khác nhau.

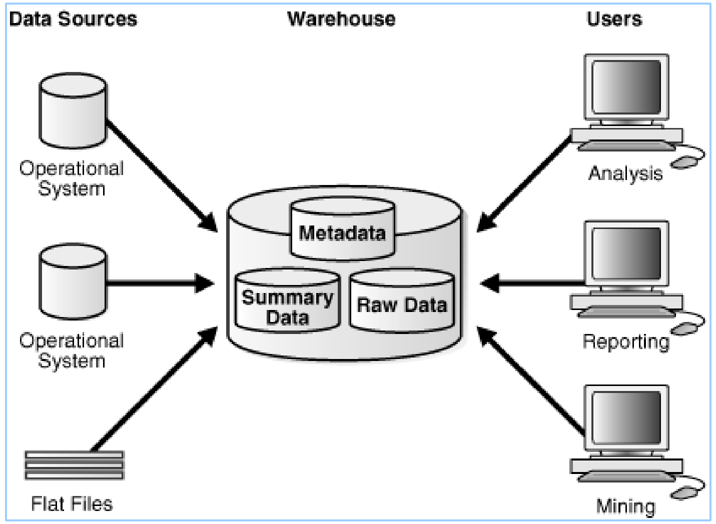
1.5. Data Warehouse Architectures

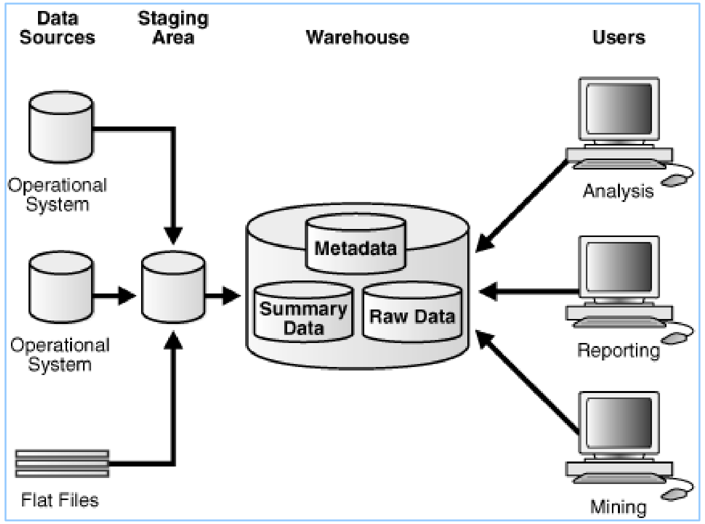
- Back-end tier
  - ETL process
  - Data Staging Area (DSA): Data was integrated and transformed then load to DW
- Data warehouse tier
  - Enterprise DW and/or several data marts
  - Metadata
- OLAP tier
  - Provides a multidimensional view
- Front-end tier
  - Used for data analysis and visualization
  - Contains client tools: OLAP tools, reporting tools, statistical tools, and data-mining tools

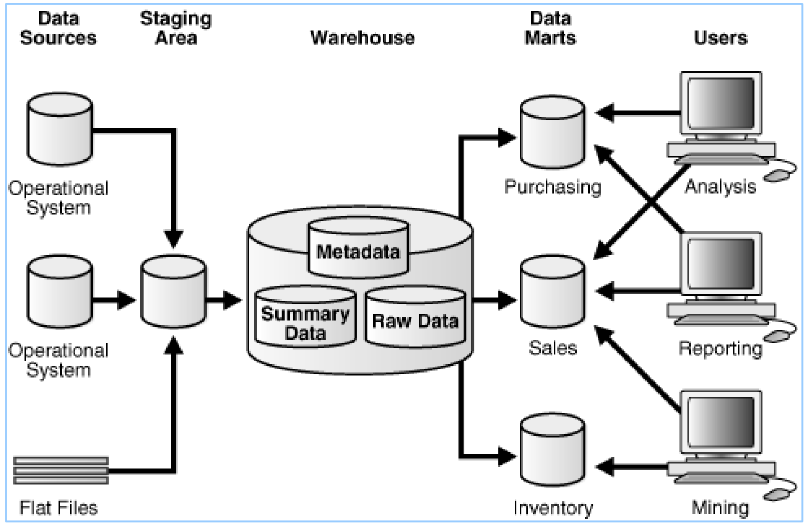


1.5. Data Warehouse Architectures

- Three common architectures
  - Data Warehouse Architecture: Basic
  - Data Warehouse Architecture: with a Staging Area
  - Data Warehouse Architecture: with a Staging Area and Data Marts



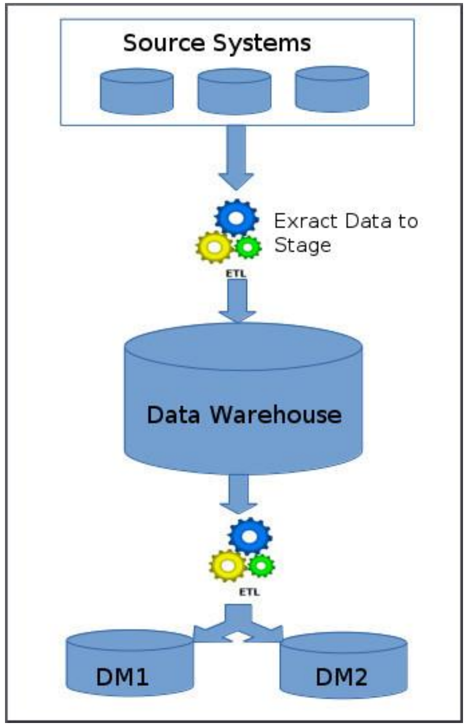


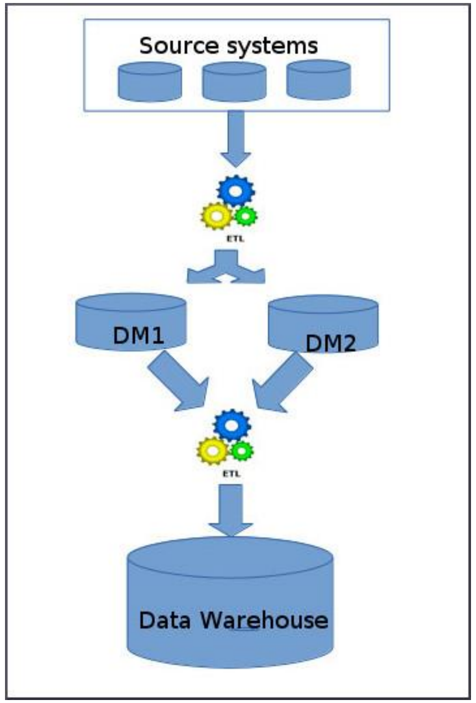


1.5. Data Warehouse Architectures

Bottom Up Vs Top Down Approach in Data Warehouse

Top-Down Approach	Bottom-Up Approach
Provides a definite and consistent view of information as information from the data warehouse is used to create Data Marts	Reports can be generated easily as Data marts are created first and it is relatively easy to interact with data marts.
Strong model and hence preferred by big companies	Not as strong but data warehouse can be extended and the number of data marts can be created
Time, Cost and Maintenance is high	Time, Cost and Maintenance are low.
Bill Inmon	Kimball





1.5. Data Warehouse Architectures

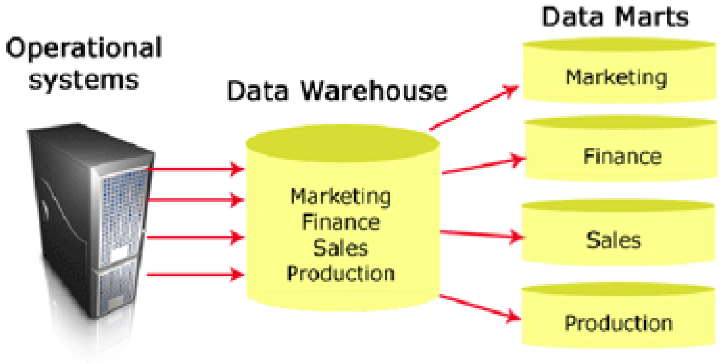
Bottom Up Vs Top Down Approach in Data Warehouse

Top-Down	Bottom-Up
Advantages	
- It is easier to maintain Top Down Design - Provides consistent dimensional views of data across data marts, as all data marts are loaded from the DW. - This approach is robust against business changes. - Creating a new data mart from the data warehouse is very easy. - Initial cost is high but subsequent project development cost is lower	- This model contains consistent data marts and these data marts can be delivered quickly. - The data marts are created first to provide reporting capability - It is easier to extend. Creating new data marts and then integrating with others. - This Approach take less time. Initial set up is very quickly
Disadvantage	
- It represents a very large project and the cost of implementing the project is significant. - It is time consuming and more time required for initial set up - Highly skilled people required for set up	- Initial cost is low but each subsequent phase will cost same - The positions of the DW and the data marts are reversed in the bottom-up approach design. - It is difficult to maintain and often redundant and subject to revisions

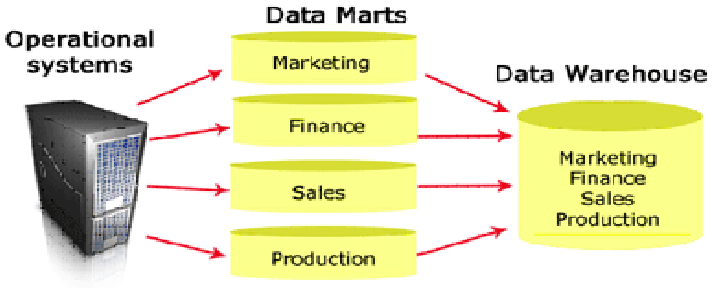
1.5. Data Warehouse Architectures

Data marts

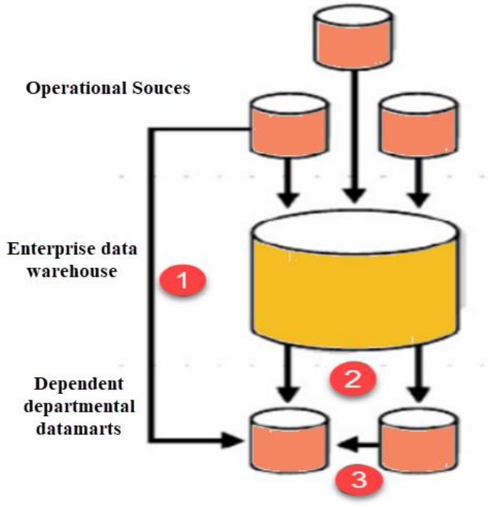
- Data Mart giúp tăng cường thời gian phản hồi của người dùng do giảm khối lượng dữ liệu.
- Chi phí triển khai Data Mart thấp hơn so với việc triển khai kho dữ liệu đầy đủ.
- Dependent Data Mart
  - Data Mart phụ thuộc chứa những dữ liệu được lấy từ Data Warehouse
  - Dữ liệu được trích lọc và tinh chế, tích hợp lại ở mức cao hơn để phục vụ một chủ đề nhất định của Data Mart.
- Independent Data Mart
  - Data Mart độc lập được xây dựng trước Data Warehouse
  - Dữ liệu được lấy trực tiếp từ các nguồn khác nhau.
- Hybrid



Dependent Data Mart



Independent Data Mart



Hybrid Data Mart

1.6. Data warehouse building process

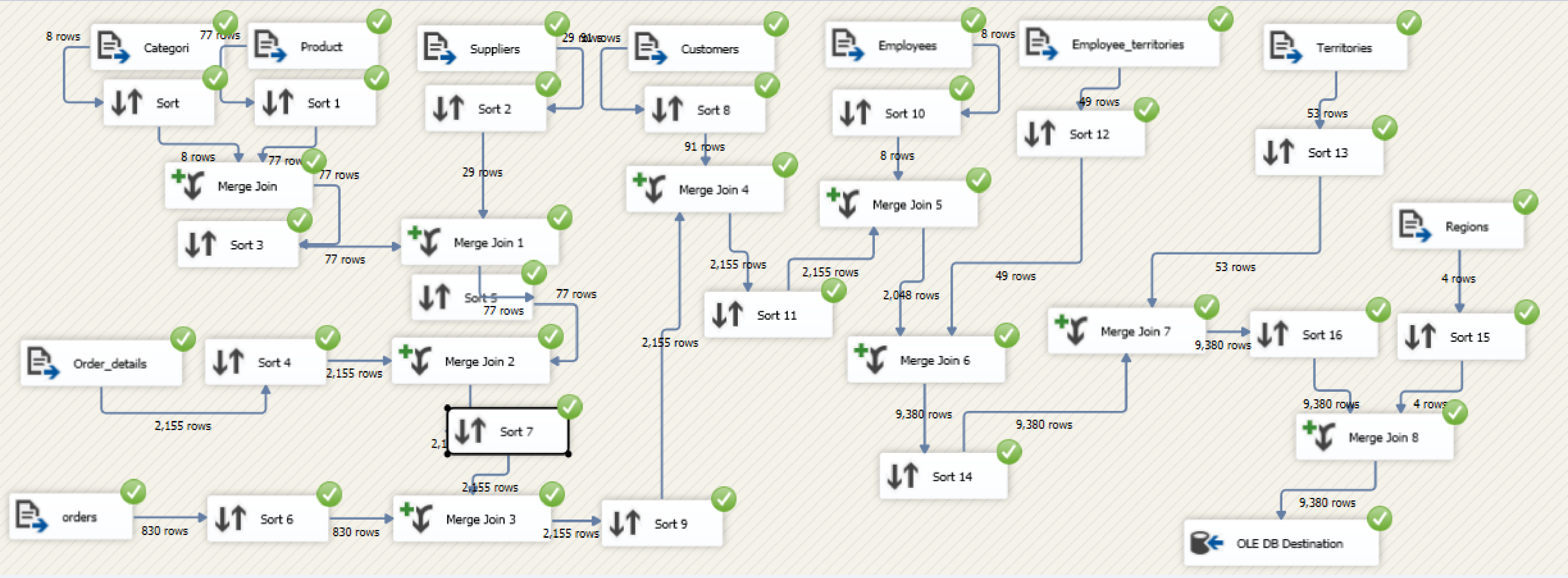
- Step 1: Determine Business Objectives and Identify Core Business Processes
- Step 2: Locate Data Sources
- Step 3: Design DW structure (Cube, Dimensions, Measures. Hierarchy…)
- Step 4: ETL (Extract, Transform, Load)
- Step 5: Implement the DW
- Step 6: Set Tracking Duration

Questions

1. What is OLTP? What are the differences between Database and Data warehouse?
2. What are the differences between architecture approaches, the architectures of data warehouse?
3. How to build a data warehouse step by step?

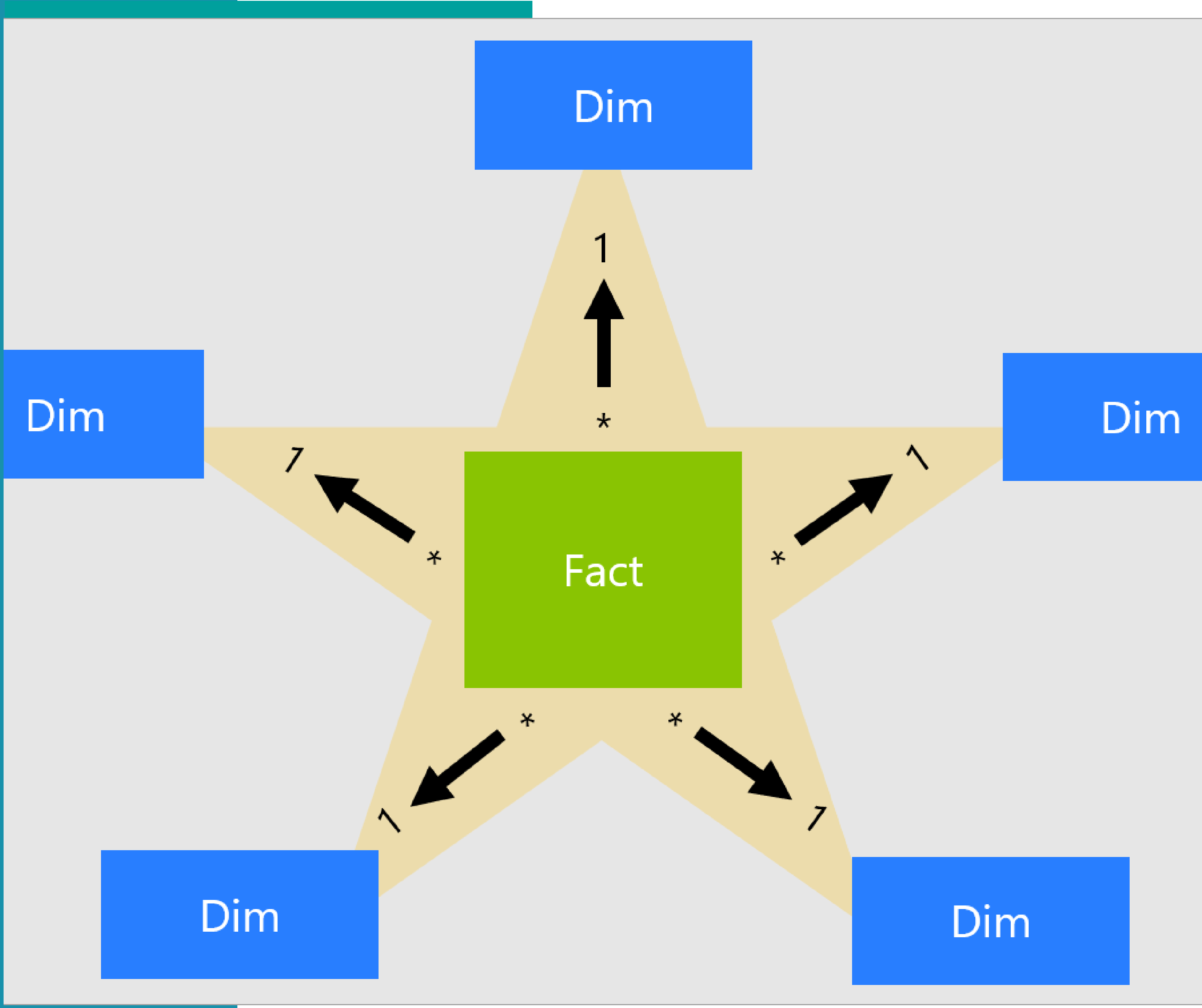
Practice

ETL (csv)



DATA WAREHOUSE

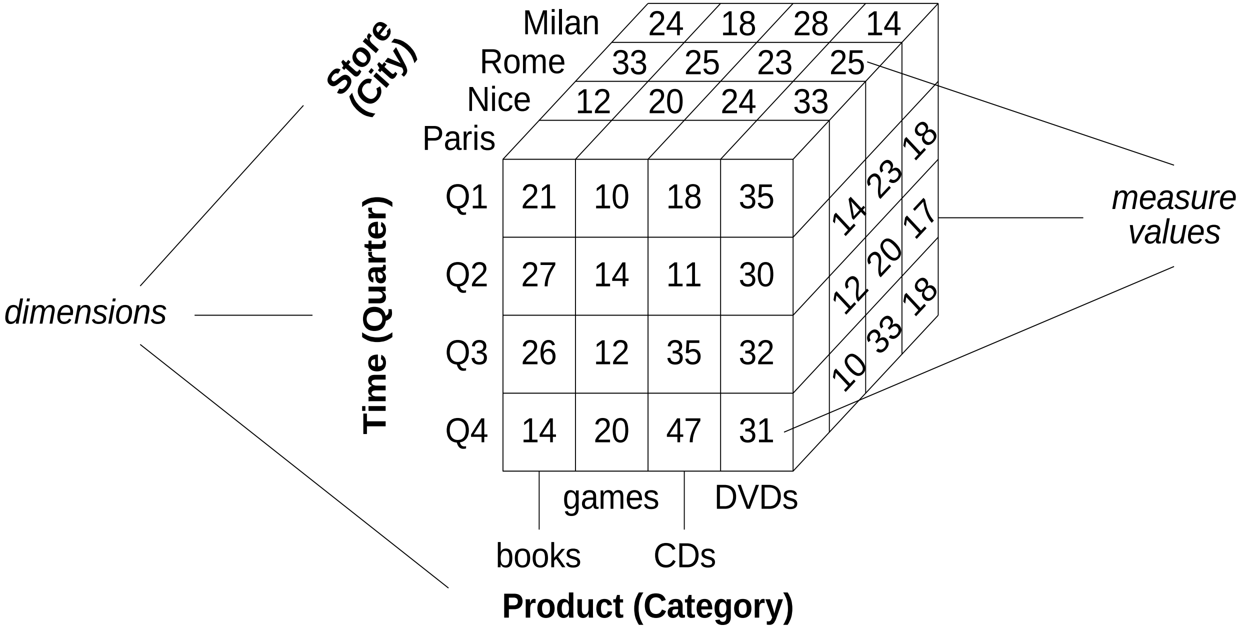
- DW vs. Online Analytical Processing (OLAP)
- Star Schema
- Snowflake Schema
- Demo ETL to Star schema



1.7. DW & Online Analytical Processing (OLAP)

Multidimensional Model

- DWs and OLAP use a multidimensional view of data
- Represented as a data cube or an hypercube
  - Dimensions: Perspectives for analyzing data
  - Cells (facts): Contain measures, values that are to be analyzed



	books	games	CDs	DVDs
Q1	21	10	18	35
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

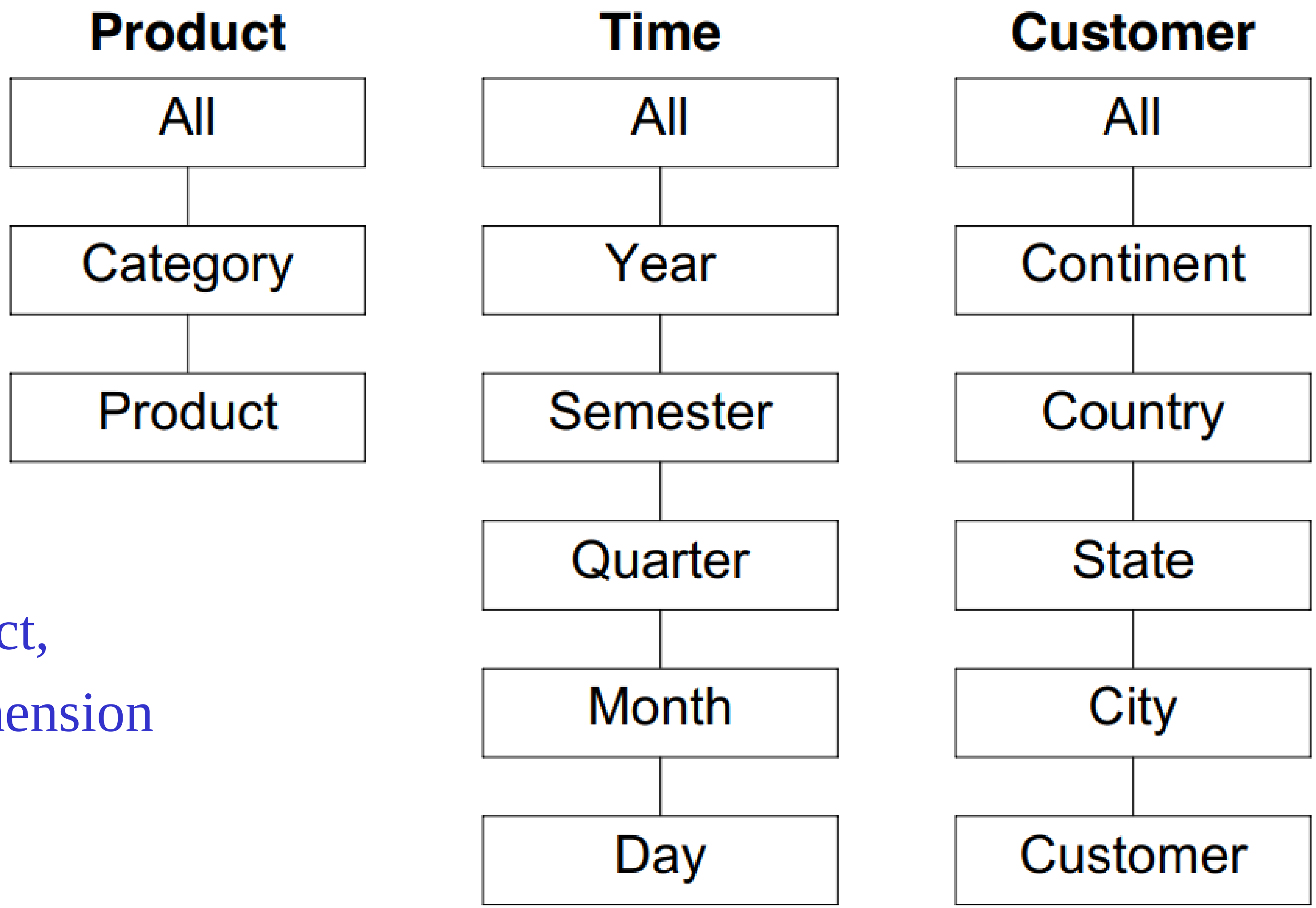
1.7. DW & Online Analytical Processing (OLAP)

Hierarchies

- Data granularity (độ mịn): Level of detail of measures
- Data analyzed at different granularities
- Hierarchies relate low-level (detailed) concepts to higher-level (general concepts)
  - Example: Store – City – Region/Province – Country
- Given two related levels in a hierarchy, lower level is called child, higher level is called parent
- Instances of these levels are called members

1.7. DW & Online Analytical Processing (OLAP)

- **Hierarchies**
  - Example
    - Hierarchies of the Product,
    - Time, and Customer dimension



Product

All

Category

Product

Time

All

Year

Semester

Quarter

Month

Day

Customer

All

Continent

Country

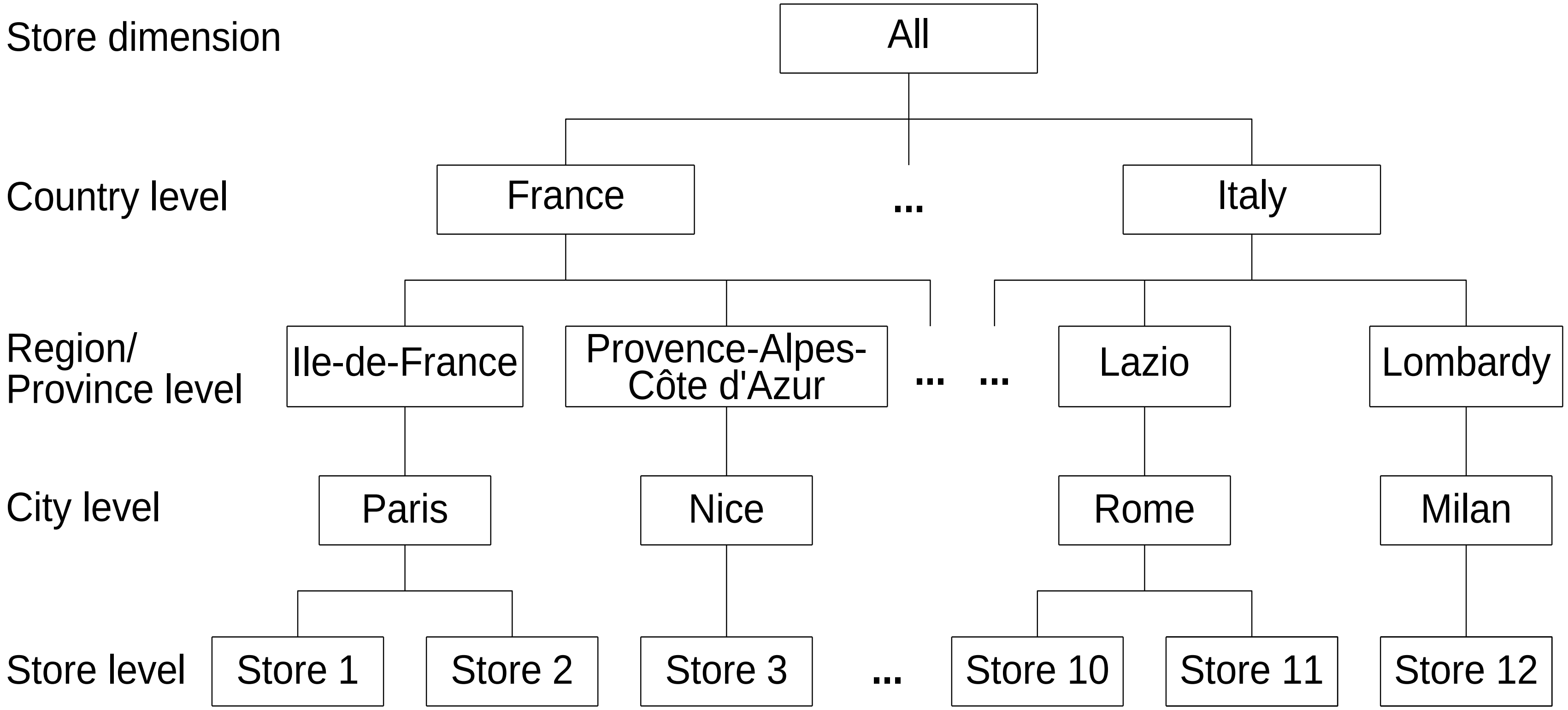
State

City

Customer

1.7. DW & Online Analytical Processing (OLAP)

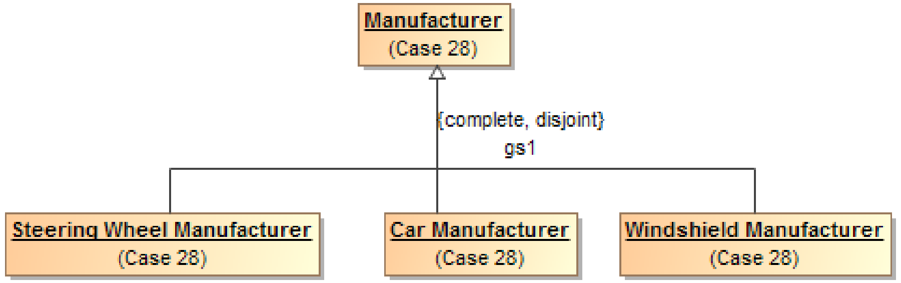
Hierarchies



1.7. DW & Online Analytical Processing (OLAP)

Measure Aggregation and Summarizability

- Measures are aggregated when using hierarchies for visualizing data at different abstraction levels
- Summarizability conditions ensure correct aggregation
  - Disjointness of instances: Grouping of instances in a level with respect to the parent in the next level must result in disjoint sets
  - Completeness: All instances are included in the hierarchy and each instance is related to one parent in the next level
  - Correct use of aggregation functions: Type of measures determine the kind of aggregation functions that can be applied



1.7. DW & Online Analytical Processing (OLAP)

Elements of Dimensional Data Model

- A data structure technique optimized for data storage in a DW
- The purpose of dimensional modeling is to optimize the database for faster retrieval of data.
- The concept of Dimensional Modelling was developed by Ralph Kimball and consists of “fact” and “dimension” tables.

1.7. DW & Online Analytical Processing (OLAP)

Elements of Dimensional Data Model

- **Fact**
  - Result from a business process or business event
    - Facts are usually numeric and additive
  - Granularity/grain (độ mịn)
    - Identifies the fact level of detail
    - One row per sale, one row per service call, one row per claim, …
    - Atomic grain is most flexible

1.7. DW & Online Analytical Processing (OLAP)

Elements of Dimensional Data Model

Dimension

- Provides the context surrounding a business process event.
- They give who, when, what, where of a fact.
  - Ex. Product, Customer, Date, Patient, Vendor,
- Each dimension row is a unique occurrence

Dimension attributes

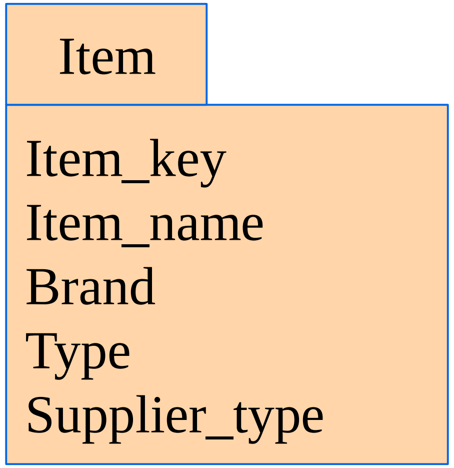
- The Attributes are the various characteristics of the dimension in dimensional data modeling.
- Ex. Location dimension: State, Country, Zip code etc
- Hierarchical relationships

1.7. DW & Online Analytical Processing (OLAP)

Elements of Dimensional Data Model

Dimension Table

- A dimension table contains dimensions of a fact.
- They are joined to fact table via a foreign key.
- Dimension tables are de-normalized tables.
- Dimensions offers descriptive characteristics of the facts
- No set limit set for given for number of dimensions
- The dimension can also contain hierarchical relationships



Item

Item_key
Item_name
Brand
Type
Supplier_type

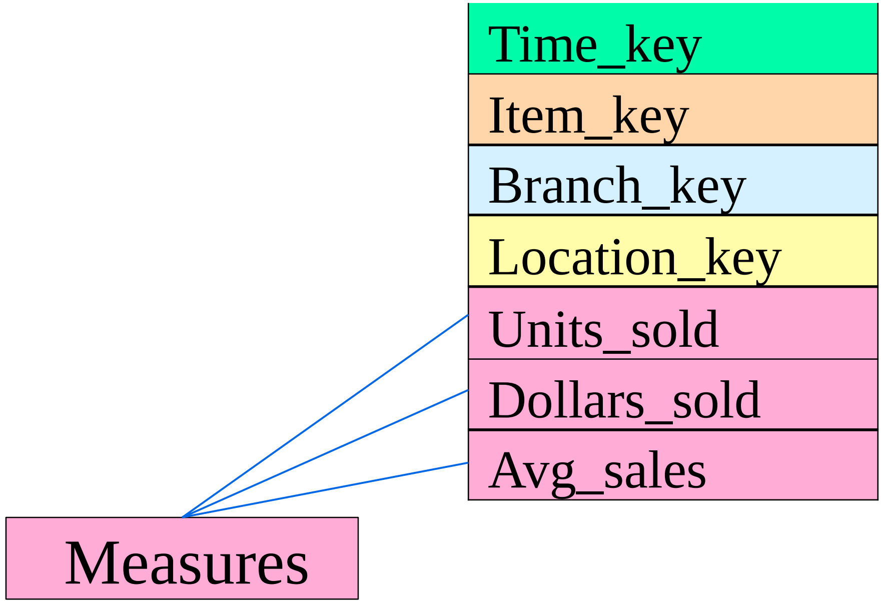
1.7. DW & Online Analytical Processing (OLAP)

Elements of Dimensional Data Model

Fact Table

- A fact table is a primary table in dimension modelling.
- A Fact Table contains
  - Measurements/facts
  - Foreign key to dimension table

Sales Fact Table



Time_key

Item_key

Branch_key

Location_key

Units_sold

Dollars_sold

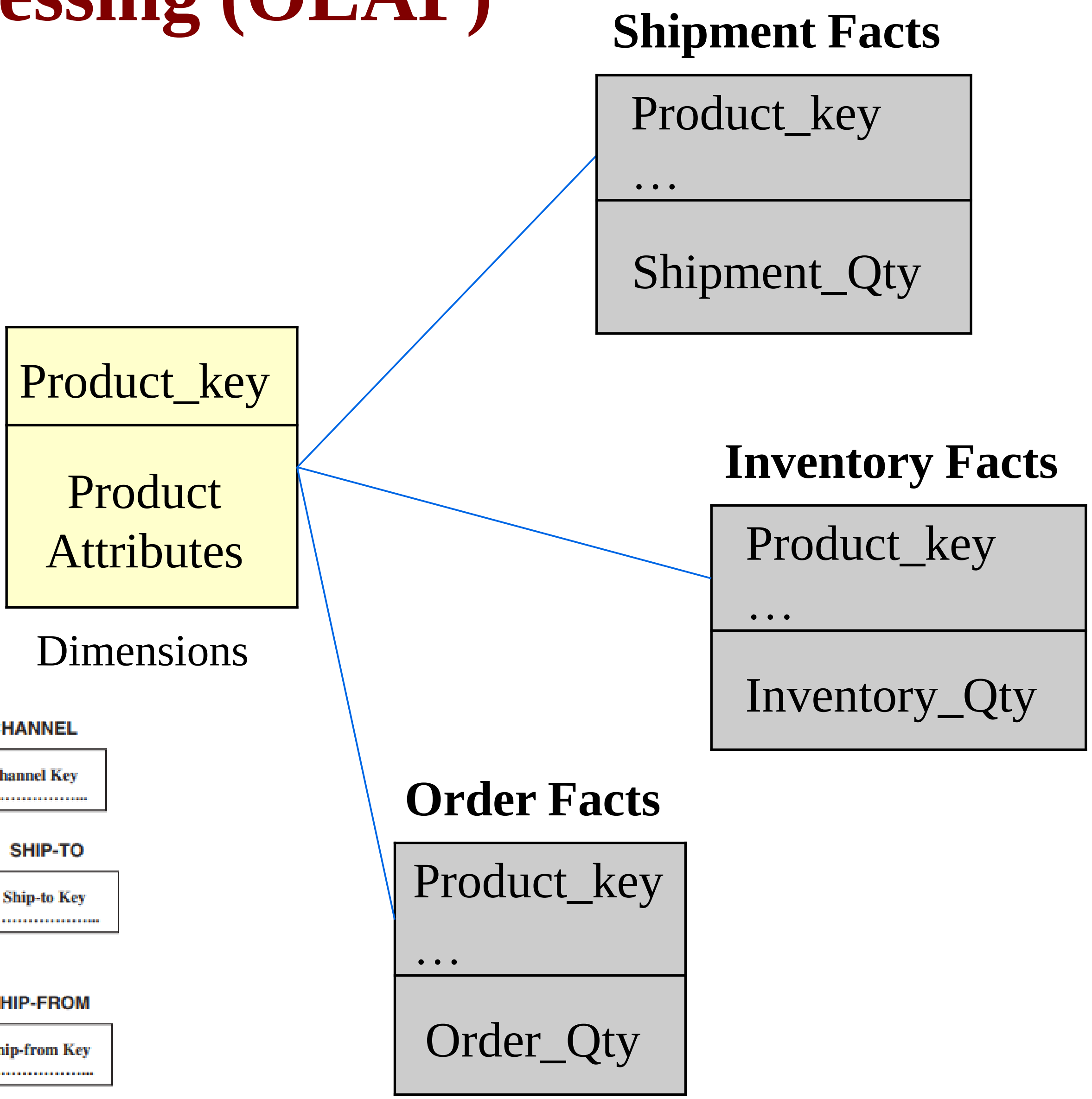
Avg_sales

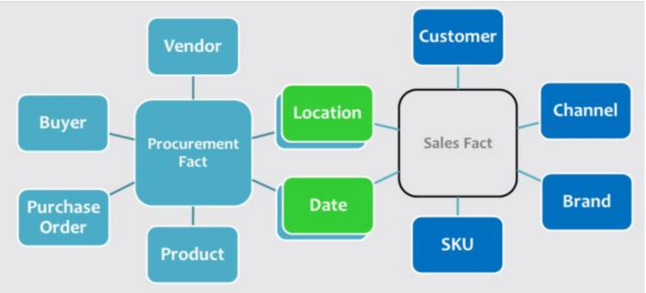
Measures

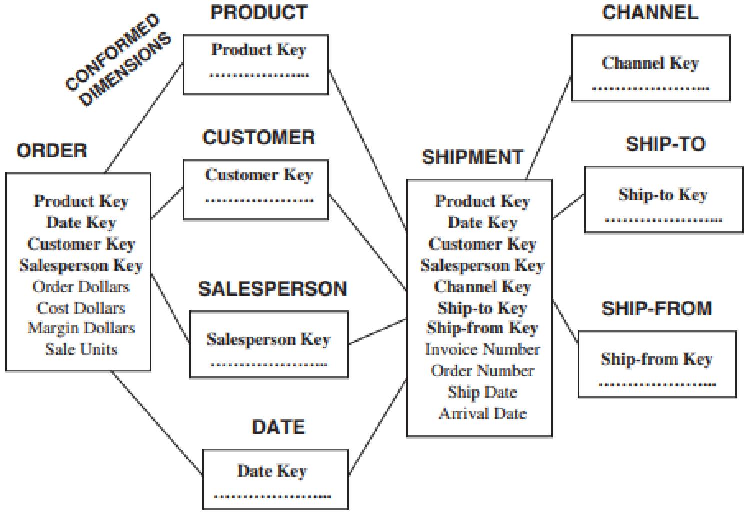
1.7. DW & Online Analytical Processing (OLAP)

Conformed Dimensions

- Shared across business processes (fact tables) in the DW
- All fact tables use same standard dimensions
- Established via Bus Matrix, enforced in ETL

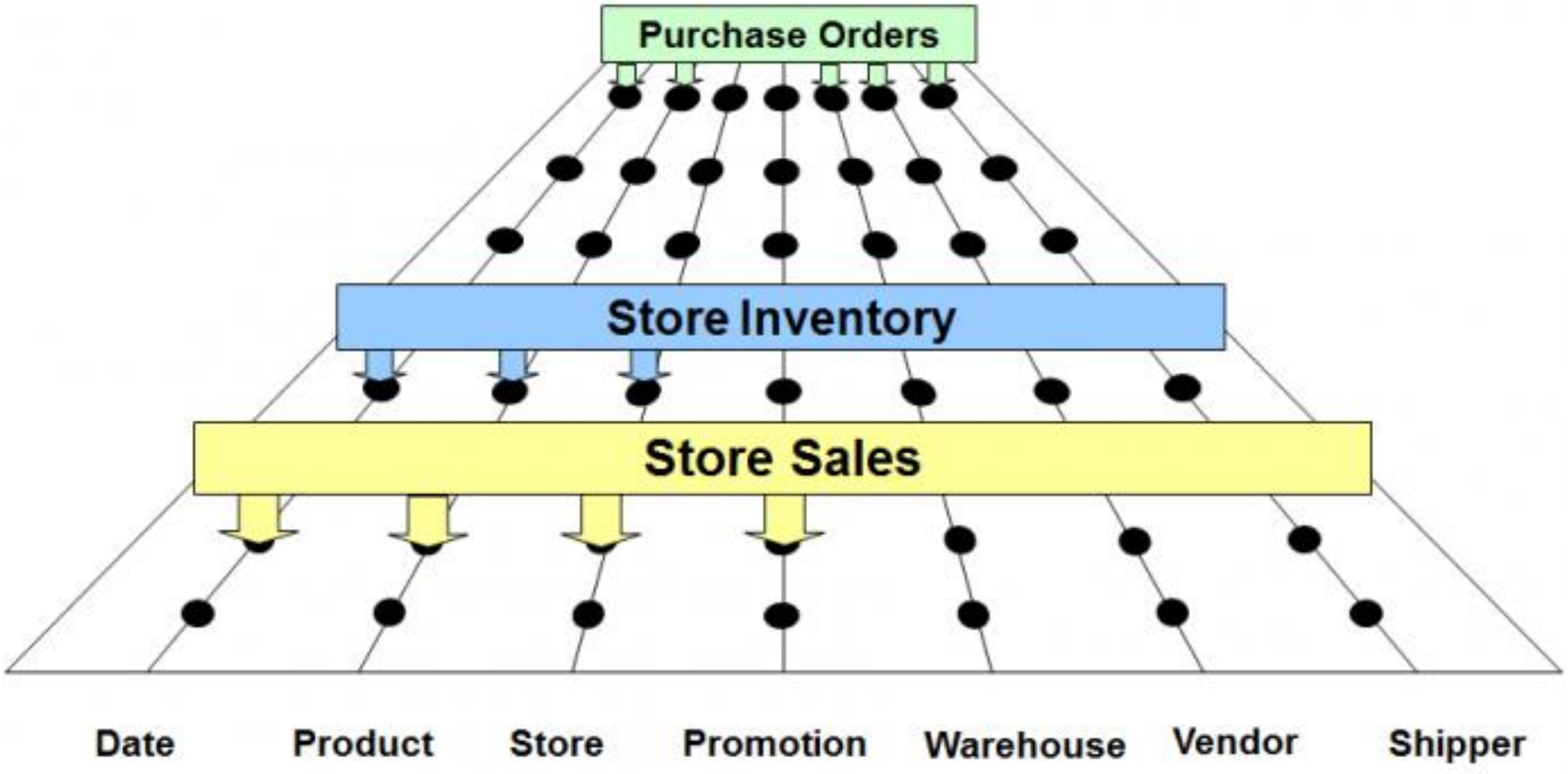






1.7. DW & Online Analytical Processing (OLAP)

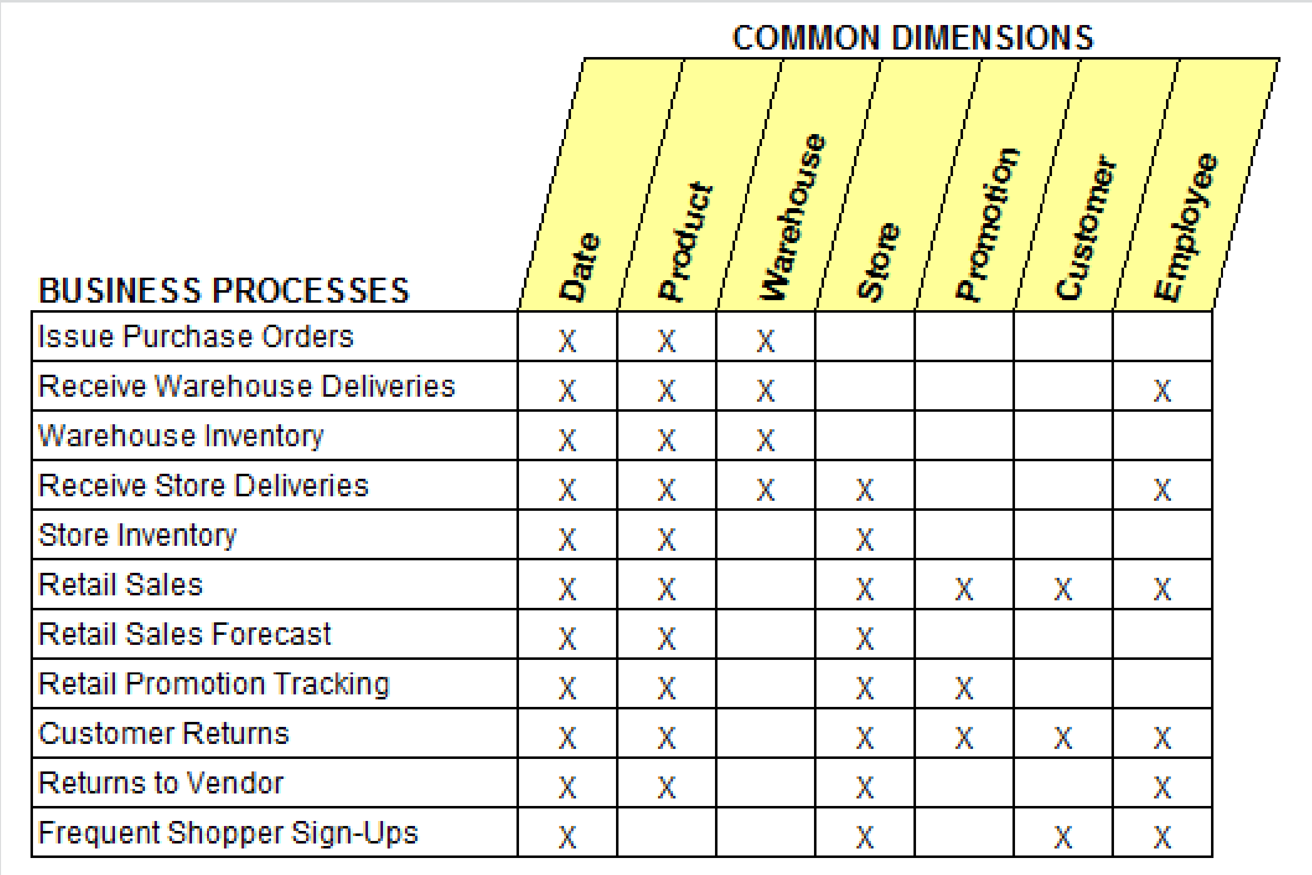
- **Enterprise DW bus Architecture**



1.7. DW & Online Analytical Processing (OLAP)

DW Bus Matrix

- Rows = Business processes
- Columns = Conformed dimensions



	COMMON DIMENSIONS						
BUSINESS PROCESSES	Date	Product	Warehouse	Store	Promotion	Customer	Employee
Issue Purchase Orders	X	X	X				
Receive Warehouse Deliveries	X	X	X				X
Warehouse Inventory	X	X	X				
Receive Store Deliveries	X	X	X	X			X
Store Inventory	X	X		X			
Retail Sales	X	X		X	X	X	X
Retail Sales Forecast	X	X		X			
Retail Promotion Tracking	X	X		X	X		
Customer Returns	X	X		X	X	X	X
Returns to Vendor	X	X		X			X
Frequent Shopper Sign-Ups	X			X		X	X

1.7. DW & Online Analytical Processing (OLAP)

Measure Classification: Additivity

- Additive measures (flow or rate measures): Can be meaningfully summarized using addition along all dimensions
  - E.g., sales amount can be summarized when the hierarchies in Store, Time, and Product dimensions are traversed
- Semi-additive measures (stock or level measures): Can be meaningfully summarized using addition along some (not all) dimensions
  - E.g., inventory quantities, can be aggregated in the Store dimension, but cannot be aggregated in the Time dimension
- Non-additive measures (value-per-unit measures): Cannot be meaningfully summarized using addition along any dimension
  - E.g., item price, cost per unit, exchange rate

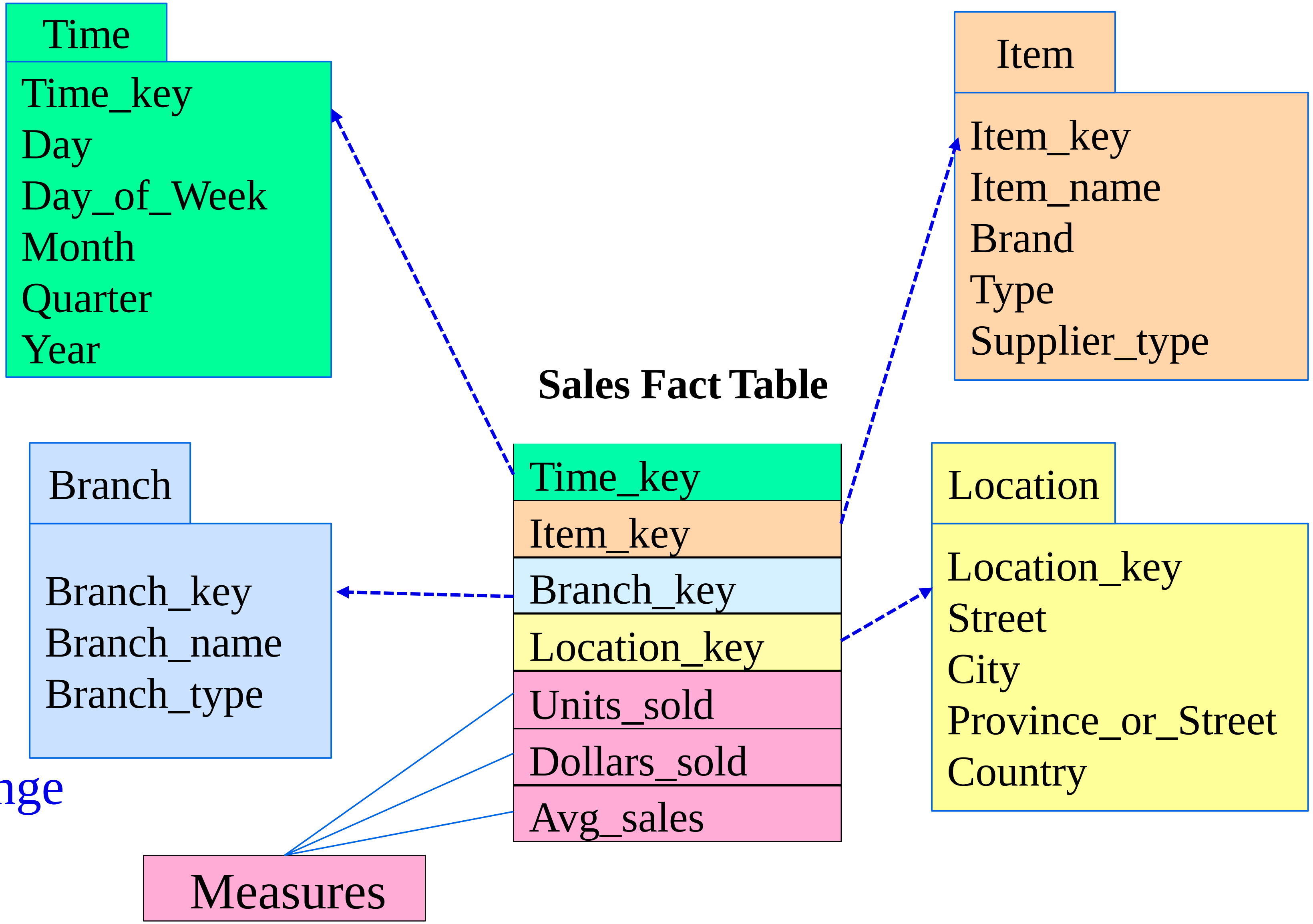
1.7. DW & Online Analytical Processing (OLAP)

Measure Classification: Aggregation Complexity

- Distributive (phân tán) measures: Defined by an aggregation function that can be computed in a distributed way
  - Data is partitioned into n sets, aggregate function applied to each set, aggregated value is computed by applying a function to these n sub-aggregate values
  - E.g., sum, min, max, count (distinct count is not)
- Algebraic (đại số) measures: Defined by an aggregation function that has can be expressed as a scalar function of distributive function
  - E.g., average (can computed by sum/count)
- Holistic (tổng thể) measures : Cannot be computed from other sub-aggregate values
  - E.g., median, mode, rank

Star Schema

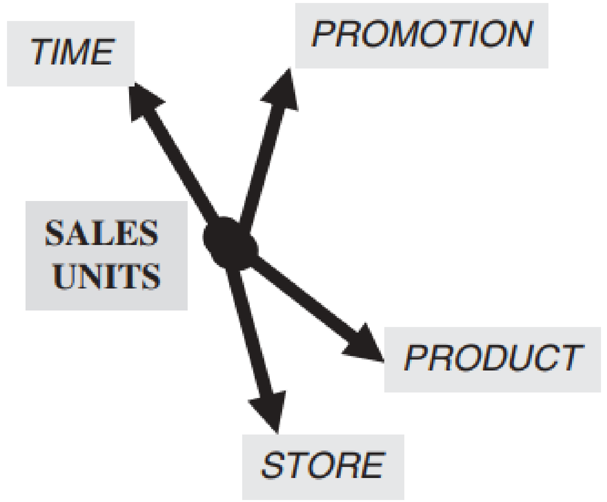
- Fact table per business process / event, plus relevant dimensions
- Benefits
  - Easier to understand
  - Better performance from fewer joins
  - Extensible to handle change



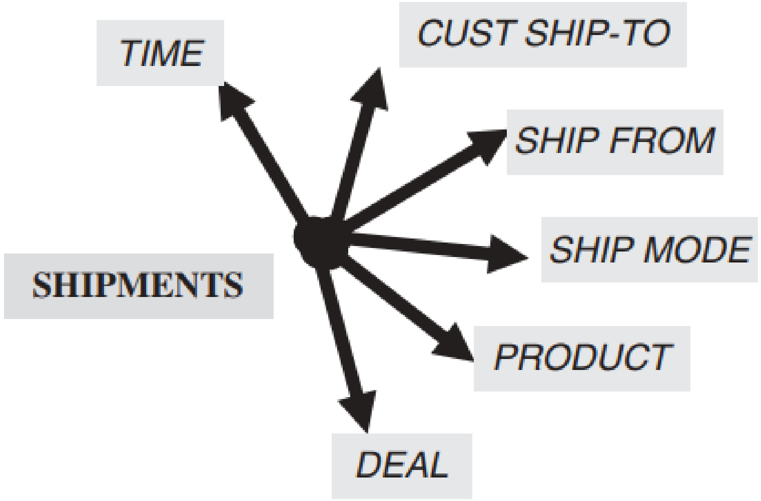
Star Schema

- Example

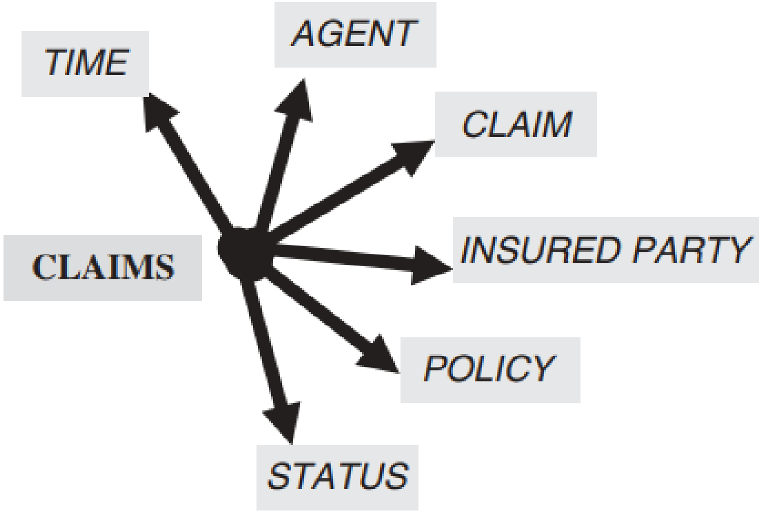
Supermarket Chain


SALES UNITS
TIME
PROMOTION
PRODUCT
STORE

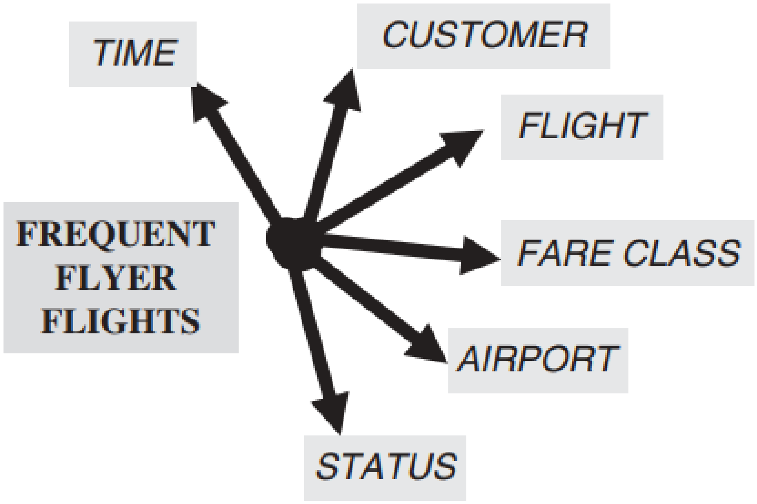
Manufacturing Company


SHIPMENTS
TIME
CUST SHIP-TO
SHIP FROM
SHIP MODE
PRODUCT
DEAL

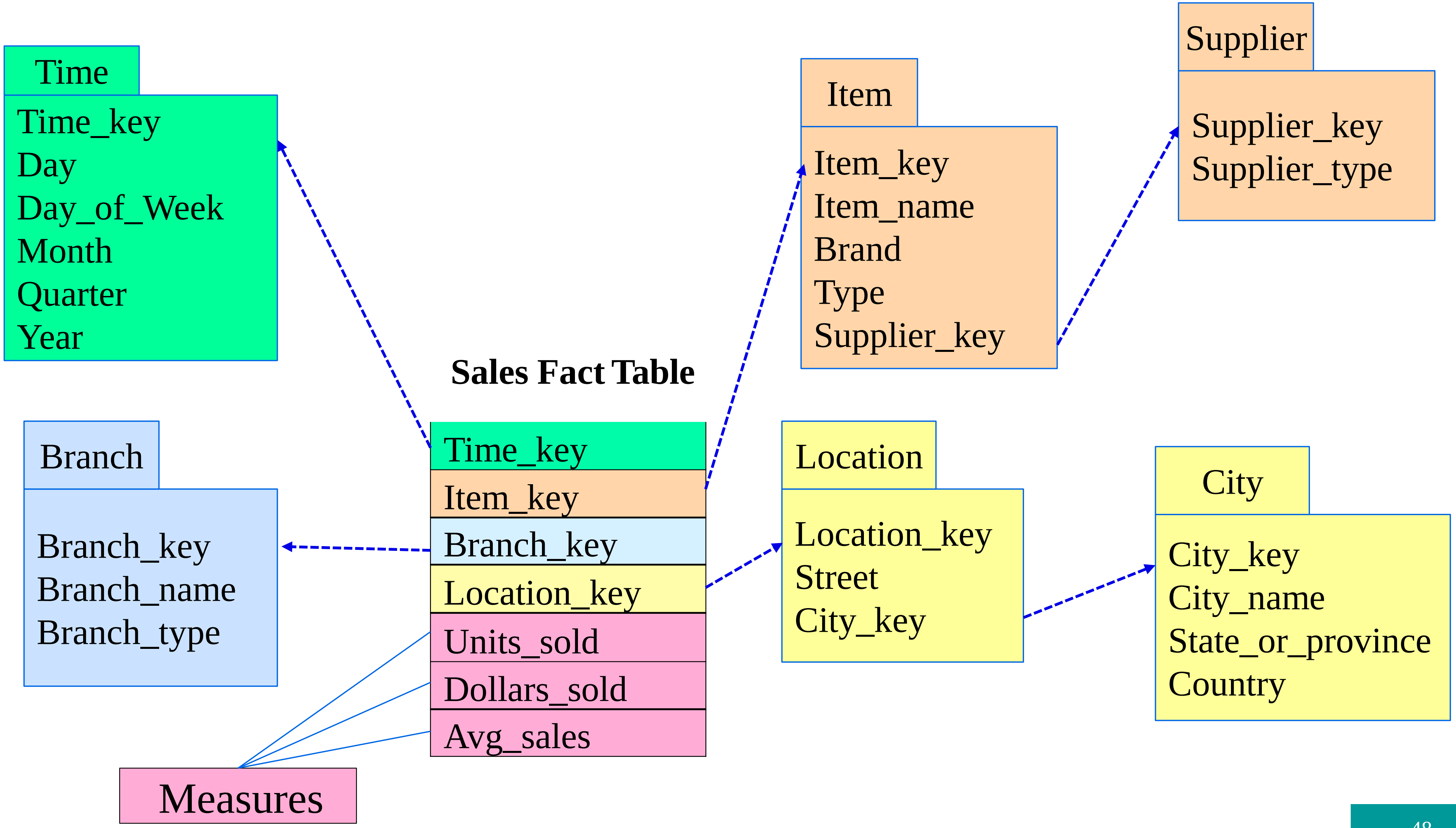
Insurance Business


CLAIMS
TIME
AGENT
CLAIM
INSURED PARTY
POLICY
STATUS

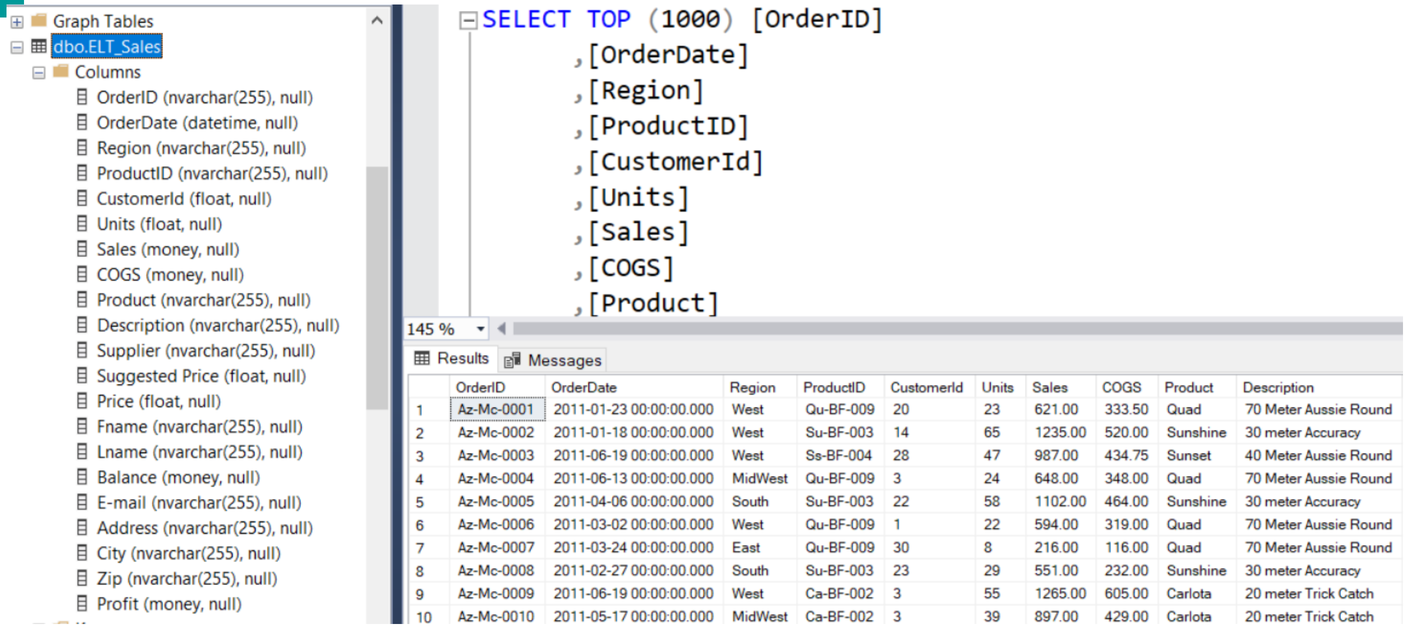
Airlines Company


FREQUENT FLYER FLIGHTS
TIME
CUSTOMER
FLIGHT
FARE CLASS
AIRPORT
STATUS

Snowflake Schema



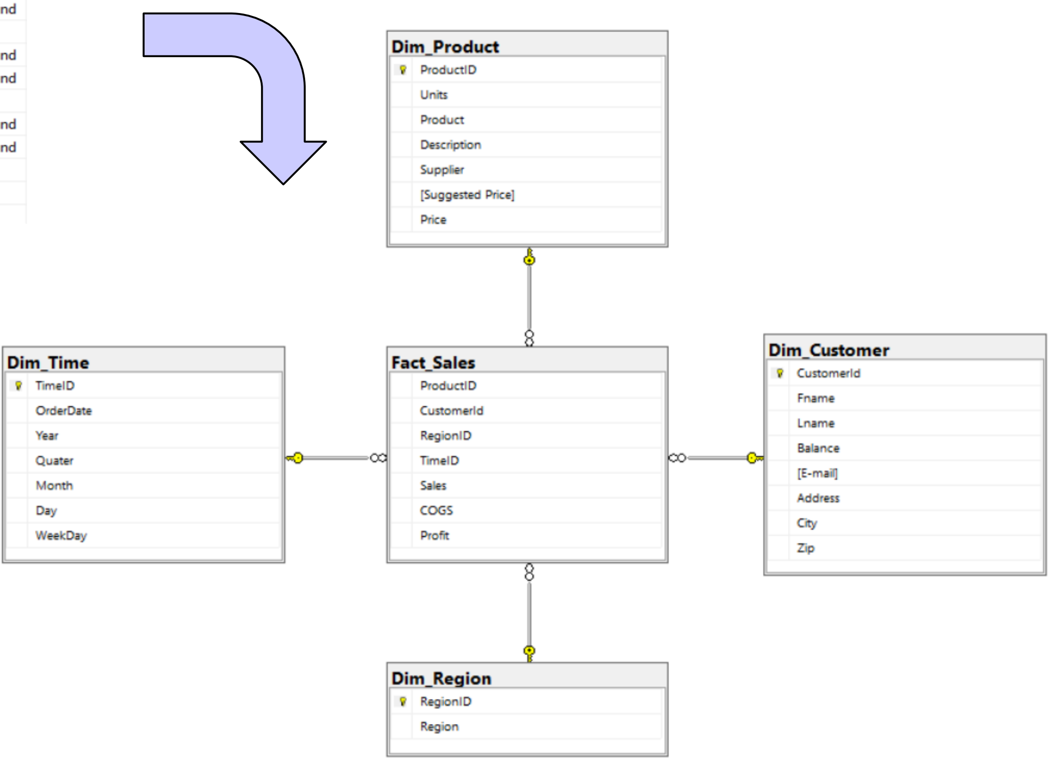
Assignment: ETL to Star schema



```
OrderID (nvarchar(255), null)
OrderDate (datetime, null)
Region (nvarchar(255), null)
ProductID (nvarchar(255), null)
CustomerId (float, null)
Units (float, null)
Sales (money, null)
COGS (money, null)
Product (nvarchar(255), null)
Description (nvarchar(255), null)
Supplier (nvarchar(255), null)
Suggested Price (float, null)
Price (float, null)
Fname (nvarchar(255), null)
Lname (nvarchar(255), null)
Balance (money, null)
E-mail (nvarchar(255), null)
Address (nvarchar(255), null)
City (nvarchar(255), null)
Zip (nvarchar(255), null)
Profit (money, null)
```

```sql
SELECT TOP (1000) [OrderID]
      ,[OrderDate]
      ,[Region]
      ,[ProductID]
      ,[CustomerId]
      ,[Units]
      ,[Sales]
      ,[COGS]
      ,[Product]
```

	OrderID	OrderDate	Region	ProductID	CustomerId	Units	Sales	COGS	Product	Description
1	Az-Mc-0001	2011-01-23 00:00:00.000	West	Qu-BF-009	20	23	621.00	333.50	Quad	70 Meter Aussie Round
2	Az-Mc-0002	2011-01-18 00:00:00.000	West	Su-BF-003	14	65	1235.00	520.00	Sunshine	30 meter Accuracy
3	Az-Mc-0003	2011-06-19 00:00:00.000	West	Ss-BF-004	28	47	987.00	434.75	Sunset	40 Meter Aussie Round
4	Az-Mc-0004	2011-06-13 00:00:00.000	MidWest	Qu-BF-009	3	24	648.00	348.00	Quad	70 Meter Aussie Round
5	Az-Mc-0005	2011-04-06 00:00:00.000	South	Su-BF-003	22	58	1102.00	464.00	Sunshine	30 meter Accuracy
6	Az-Mc-0006	2011-03-02 00:00:00.000	West	Qu-BF-009	1	22	594.00	319.00	Quad	70 Meter Aussie Round
7	Az-Mc-0007	2011-03-24 00:00:00.000	East	Qu-BF-009	30	8	216.00	116.00	Quad	70 Meter Aussie Round
8	Az-Mc-0008	2011-02-27 00:00:00.000	South	Su-BF-003	23	29	551.00	232.00	Sunshine	30 meter Accuracy
9	Az-Mc-0009	2011-06-19 00:00:00.000	West	Ca-BF-002	3	55	1265.00	605.00	Carlota	20 meter Trick Catch
10	Az-Mc-0010	2011-05-17 00:00:00.000	MidWest	Ca-BF-002	3	39	897.00	429.00	Carlota	20 meter Trick Catch



Dim_Product
ProductID
Units
Product
Description
Supplier
[Suggested Price]
Price

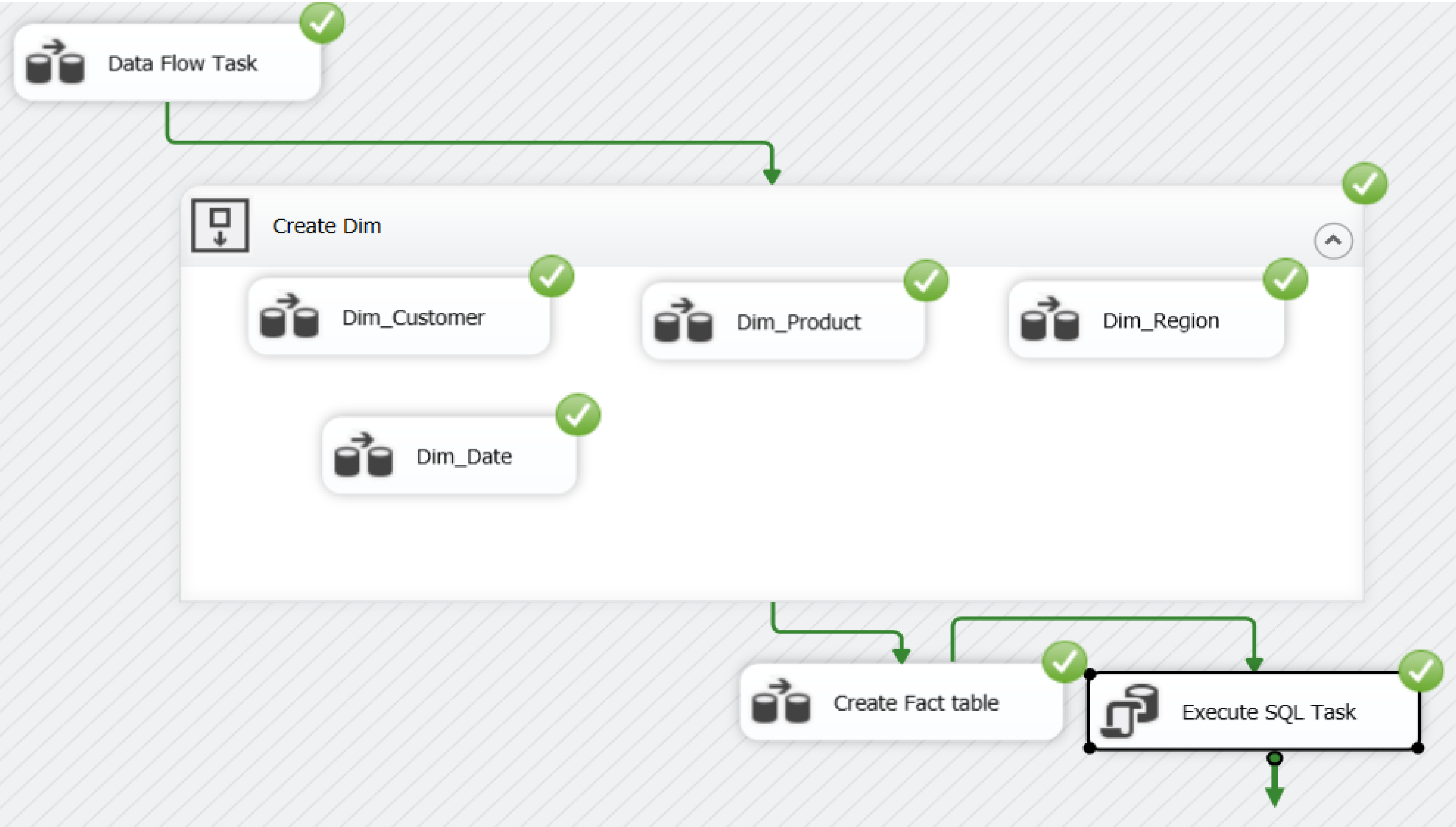
Dim_Time
TimeID
OrderDate
Year
Quater
Month
Day
WeekDay

Fact_Sales
ProductID
CustomerId
RegionID
TimeID
Sales
COGS
Profit

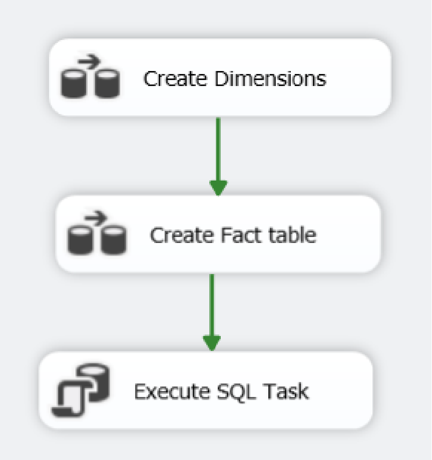
Dim_Customer
CustomerId
Fname
Lname
Balance
[E-mail]
Address
City
Zip

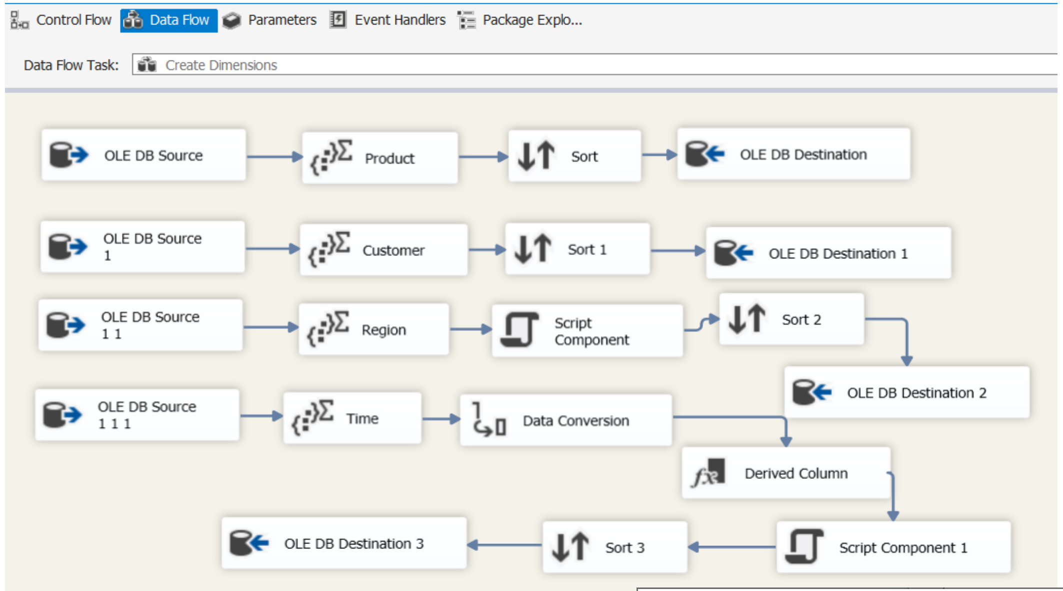
Dim_Region
RegionID
Region

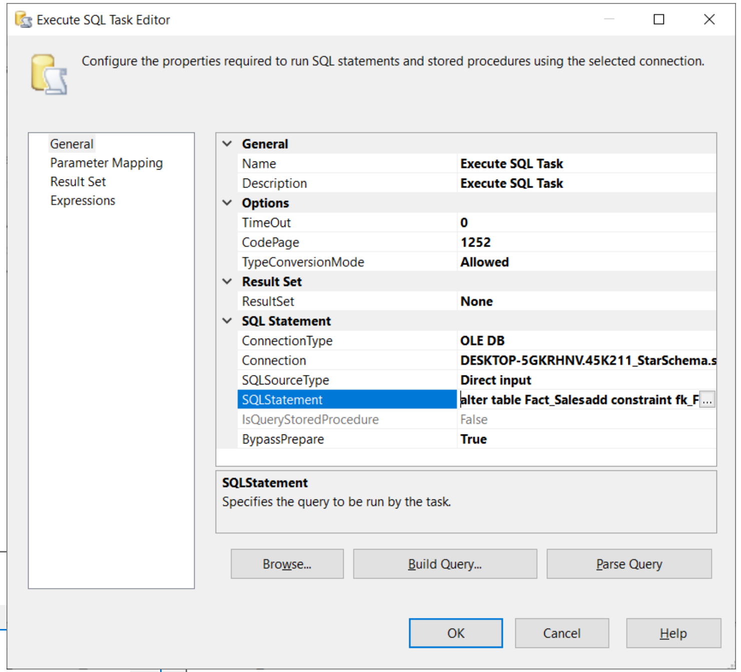
Assignment: ETL to Star schema



Practice (ELT to Star Schema)



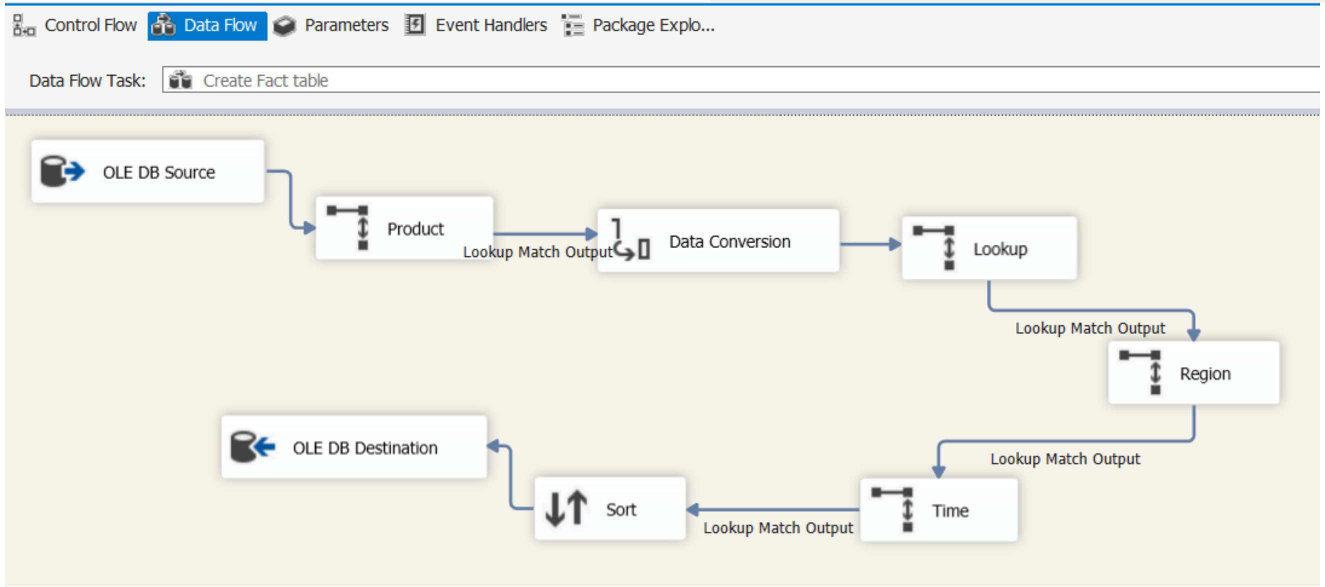




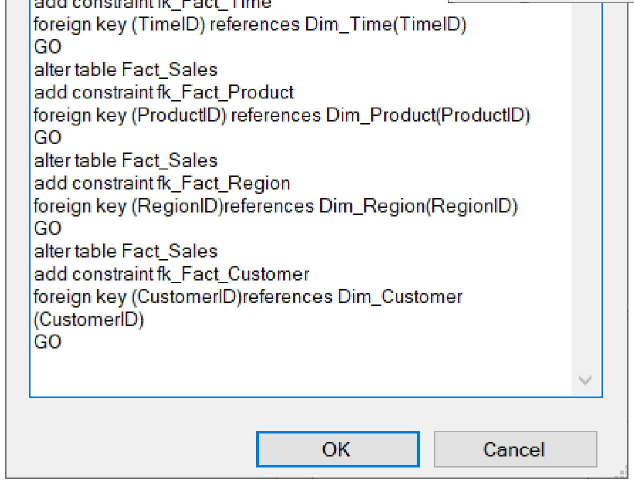
Configure the properties required to run SQL statements and stored procedures using the selected connection.

General	
Name	Execute SQL Task
Description	Execute SQL Task
Options	
TimeOut	0
CodePage	1252
TypeConversionMode	Allowed
Result Set	
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	DESKTOP-5GKRHNV.45K211_StarSchema.s
SQLSourceType	Direct input
SQLStatement	alter table Fact_Salesadd constraint fk_F...
IsQueryStoredProcedure	False
BypassPrepare	True

SQLStatement
Specifies the query to be run by the task.



Enter SQL Query



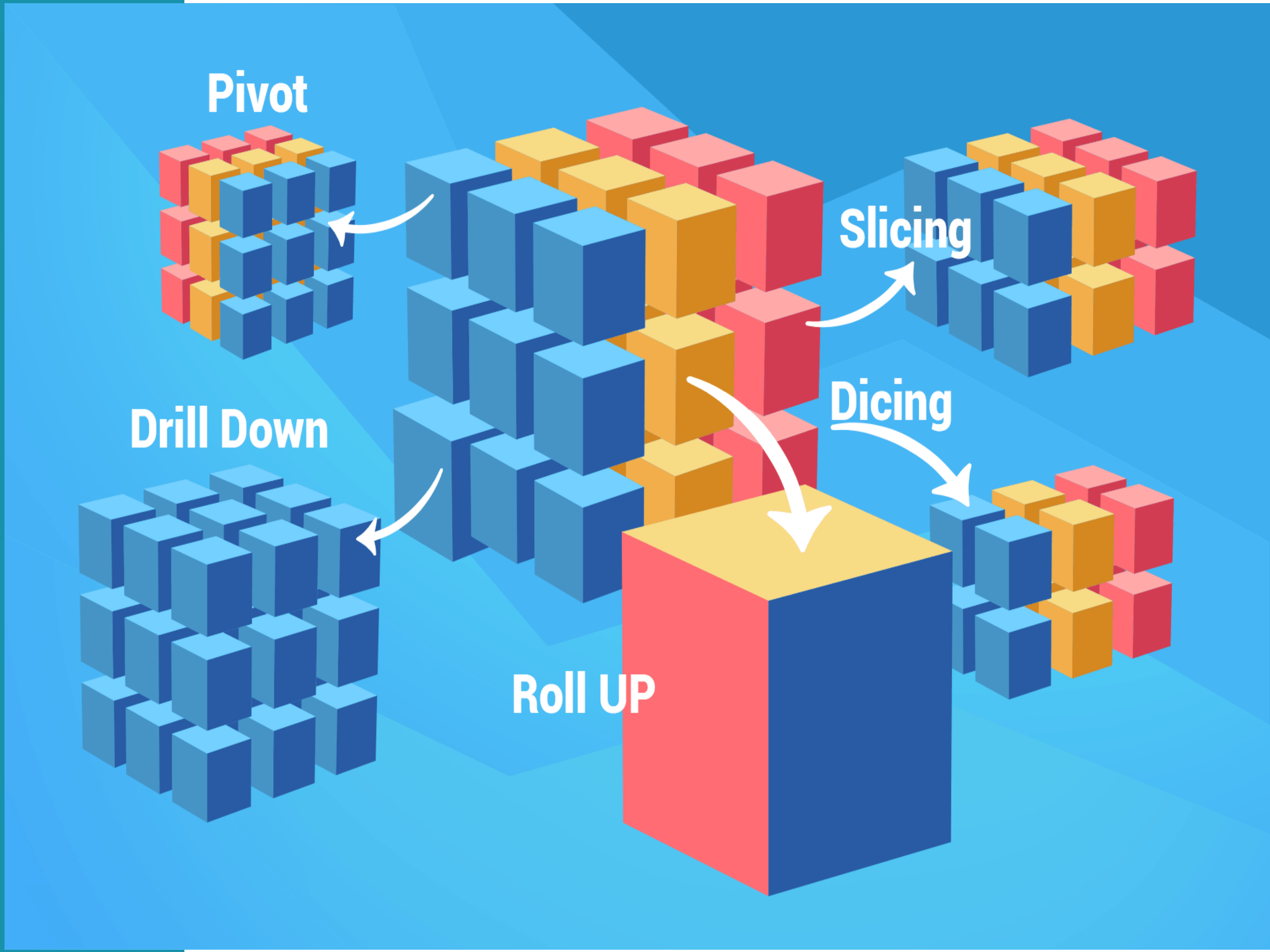
```
alter table Fact_Sales
add constraint fk_Fact_Time
foreign key (TimeID) references Dim_Time(TimeID)
GO
alter table Fact_Sales
add constraint fk_Fact_Product
foreign key (ProductID) references Dim_Product(ProductID)
GO
alter table Fact_Sales
add constraint fk_Fact_Region
foreign key (RegionID)references Dim_Region(RegionID)
GO
alter table Fact_Sales
add constraint fk_Fact_Customer
foreign key (CustomerID)references Dim_Customer
(CustomerID)
GO
```

Questions

1. What is a multidimensional model, data cube, hierarchies?
2. What are elements of Dimensional Model, conformed dimensions, enterprise DW bus architecture, DW bus matrix?
3. What are facts, measures, measure classification?
4. What is Star schema, fact table, dimension table, Snowflake Schema?

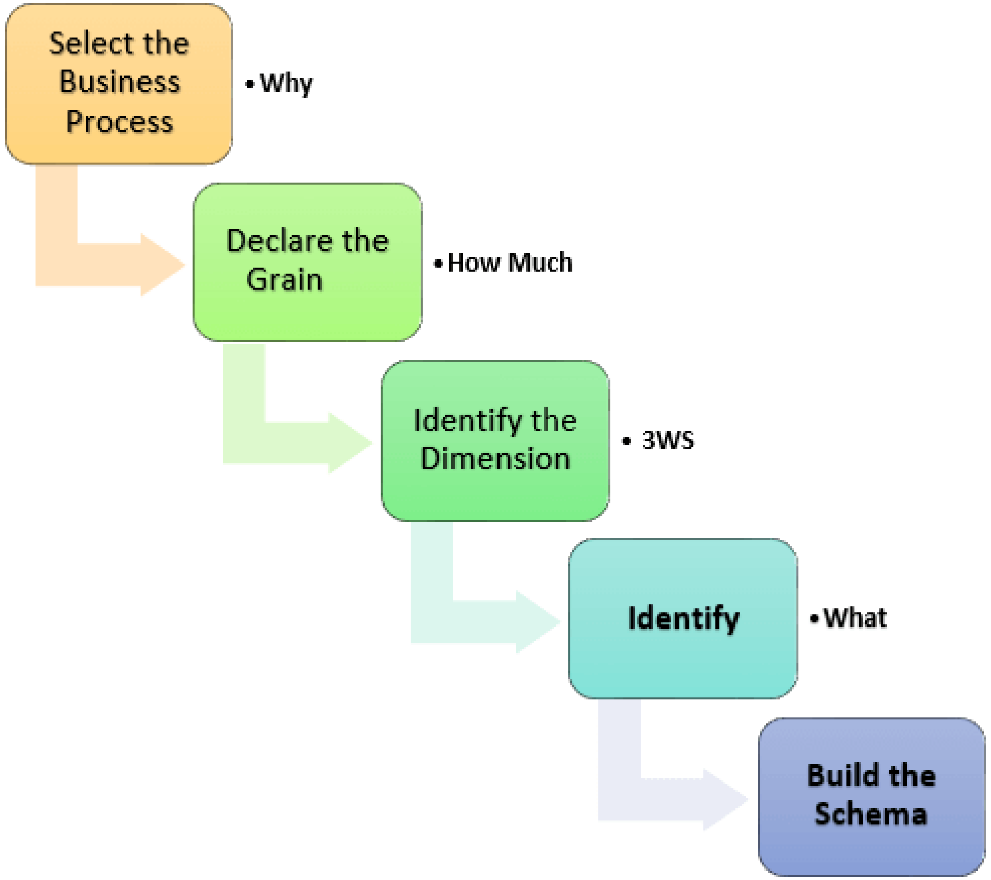
DATA WAREHOUSE

- Steps to Create dimensional modelling
- OLAP operations
- Slowly changing dimensions
- Rapidly changing dimensions
- Demo OLAP



Steps to Create Dimensional Modelling

- The accuracy in creating your Dimensional modelling determines the success of your data warehouse implementation.
  1. Identify Business Process
  2. Identify Grain (độ mịn) (level of detail)
  3. Identify Dimensions
  4. Identify Facts
  5. Build Schema



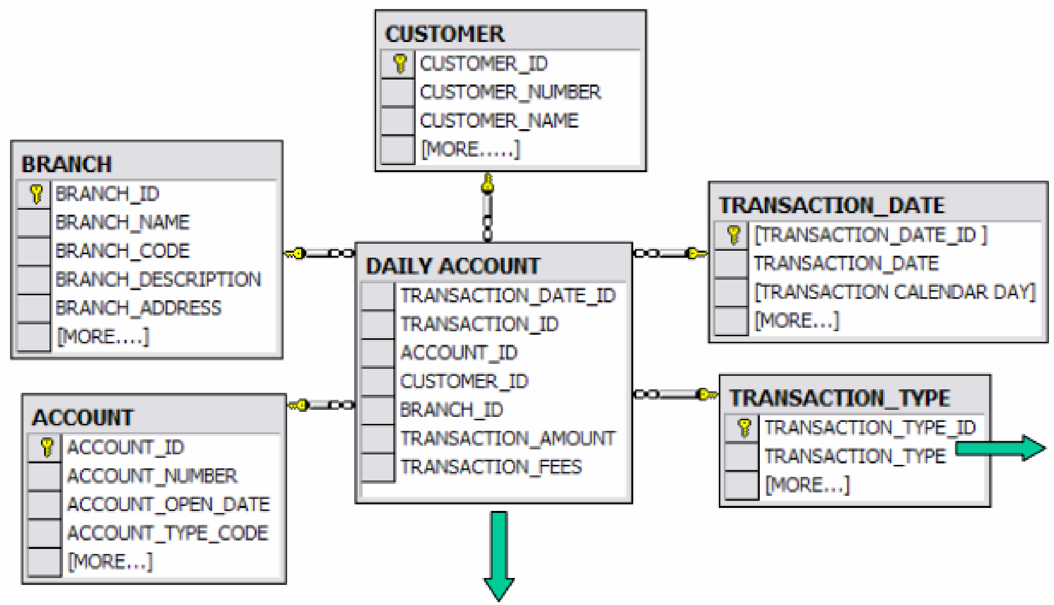
- The model should describe
  - Why, How much, When/Where/Who
  - What of your business process

Steps to Create Dimensional Modelling

- Identify Business Process
  - Identifying the actual business process a DW should cover.
    - Marketing, Sales, HR, etc.
    - Depends on the quality of data available for that process.
  - It is the most important step of the Data Modelling process
  - Type of business processes
    - Transaction
    - Accumulating Snapshot
    - Periodic Snapshot

Steps to Create Dimensional Modelling

- Identify Business Process
  - **Transactions processes**
    - The most basic fact grain
    - One row is a transaction
    - Ex. Sales, Return...



CUSTOMER
- CUSTOMER_ID
- CUSTOMER_NUMBER
- CUSTOMER_NAME
- [MORE.....]

BRANCH
- BRANCH_ID
- BRANCH_NAME
- BRANCH_CODE
- BRANCH_DESCRIPTION
- BRANCH_ADDRESS
- [MORE....]

DAILY ACCOUNT
- TRANSACTION_DATE_ID
- TRANSACTION_ID
- ACCOUNT_ID
- CUSTOMER_ID
- BRANCH_ID
- TRANSACTION_AMOUNT
- TRANSACTION_FEES

TRANSACTION_DATE
- [TRANSACTION_DATE_ID]
- TRANSACTION_DATE
- [TRANSACTION CALENDAR DAY]
- [MORE...]

ACCOUNT
- ACCOUNT_ID
- ACCOUNT_NUMBER
- ACCOUNT_OPEN_DATE
- ACCOUNT_TYPE_CODE
- [MORE...]

TRANSACTION_TYPE
- TRANSACTION_TYPE_ID
- TRANSACTION_TYPE
- [MORE...]

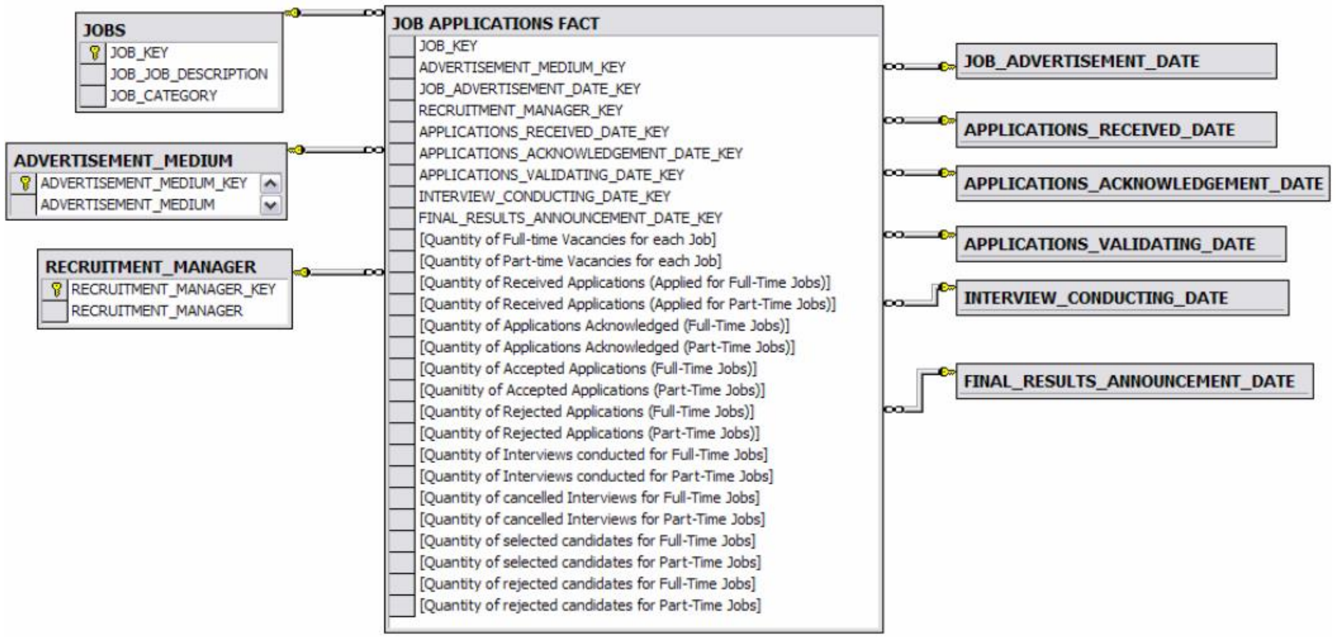
(Deposit or Withdrawal)
Transaction Type

6 Rows inserted (1 row for each Transaction)

Row 1: Withdrawal: $400, Date: 2nd August 2,2005, Time: 4:00AM
Row 2: Deposit: $300, Date: 4th August 4,2005, Time: 3:00AM
Row 3: Withdrawal: $600, Date: 5nd August 5,2005, Time: 2:00PM
Row 4: Withdrawal: $900, Date: 6th August 6,2005, Time: 9:00PM
Row 5: Deposit: $900, Date: 18th August 18,2005, Time: 7:00AM
Row 6 :Deposit: $800, Date: 23rd August 23,2005, Time: 1:00AM

Steps to Create Dimensional Modelling

- Identify Business Process
  - **Accumulating Snapshot**
    - Capture a business process workflow
    - Fact row is initially inserted, then updated as milestones (mốc) occur
    - Ex. Order fulfillment, Job application tracking…



JOBS
- JOB_KEY
- JOB_JOB_DESCRIPTION
- JOB_CATEGORY

ADVERTISEMENT_MEDIUM
- ADVERTISEMENT_MEDIUM_KEY
- ADVERTISEMENT_MEDIUM

RECRUITMENT_MANAGER
- RECRUITMENT_MANAGER_KEY
- RECRUITMENT_MANAGER

JOB APPLICATIONS FACT
- JOB_KEY
- ADVERTISEMENT_MEDIUM_KEY
- JOB_ADVERTISEMENT_DATE_KEY
- RECRUITMENT_MANAGER_KEY
- APPLICATIONS_RECEIVED_DATE_KEY
- APPLICATIONS_ACKNOWLEDGEMENT_DATE_KEY
- APPLICATIONS_VALIDATING_DATE_KEY
- INTERVIEW_CONDUCTING_DATE_KEY
- FINAL_RESULTS_ANNOUNCEMENT_DATE_KEY
- [Quantity of Full-time Vacancies for each Job]
- [Quantity of Part-time Vacancies for each Job]
- [Quantity of Received Applications (Applied for Full-Time Jobs)]
- [Quantity of Received Applications (Applied for Part-Time Jobs)]
- [Quantity of Applications Acknowledged (Full-Time Jobs)]
- [Quantity of Applications Acknowledged (Part-Time Jobs)]
- [Quantity of Accepted Applications (Full-Time Jobs)]
- [Quanitity of Accepted Applications (Part-Time Jobs)]
- [Quantity of Rejected Applications (Full-Time Jobs)]
- [Quantity of Rejected Applications (Part-Time Jobs)]
- [Quantity of Interviews conducted for Full-Time Jobs]
- [Quantity of Interviews conducted for Part-Time Jobs]
- [Quantity of cancelled Interviews for Full-Time Jobs]
- [Quantity of cancelled Interviews for Part-Time Jobs]
- [Quantity of selected candidates for Full-Time Jobs]
- [Quantity of selected candidates for Part-Time Jobs]
- [Quantity of rejected candidates for Full-Time Jobs]
- [Quantity of rejected candidates for Part-Time Jobs]

JOB_ADVERTISEMENT_DATE

APPLICATIONS_RECEIVED_DATE

APPLICATIONS_ACKNOWLEDGEMENT_DATE

APPLICATIONS_VALIDATING_DATE

INTERVIEW_CONDUCTING_DATE

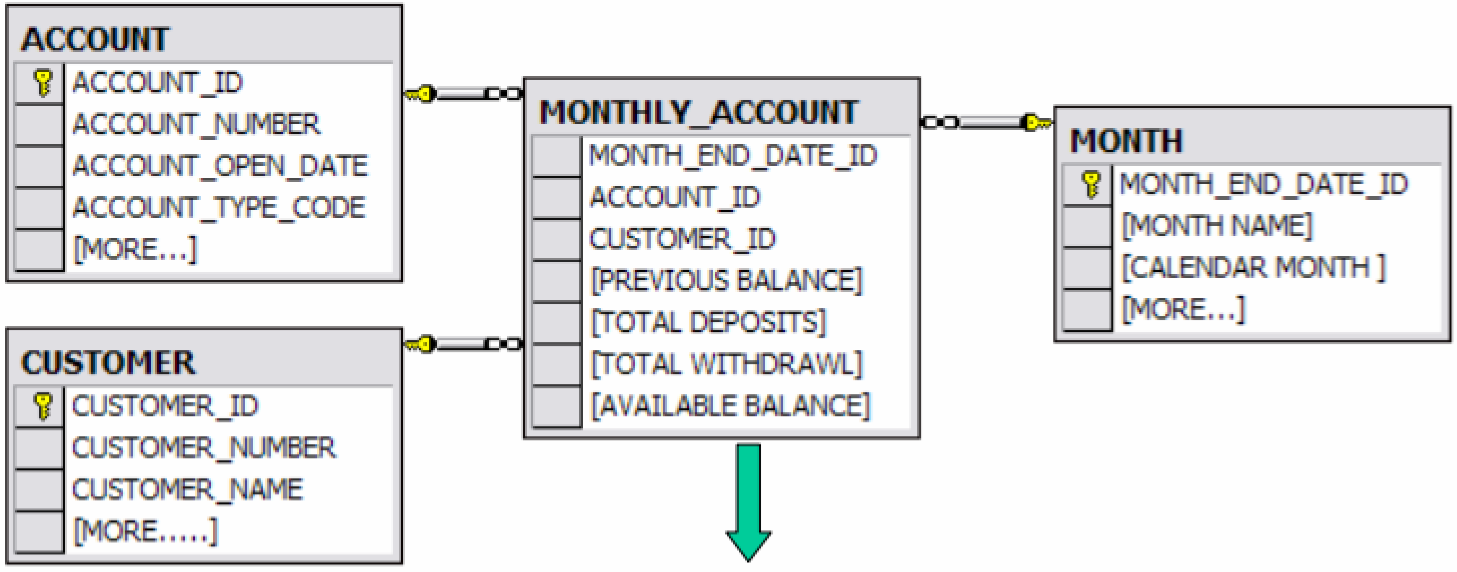
FINAL_RESULTS_ANNOUNCEMENT_DATE

Steps to Create Dimensional Modelling

- Identify Business Process

Periodic Snapshot

- At predetermined intervals snapshots of the same level of details are taken and stacked consecutively in the fact table
- Snapshots can be taken daily, weekly, monthly…
- Ex. Financial reports, Bank account values, GPA…



ACCOUNT
- ACCOUNT_ID
- ACCOUNT_NUMBER
- ACCOUNT_OPEN_DATE
- ACCOUNT_TYPE_CODE
- [MORE…]

CUSTOMER
- CUSTOMER_ID
- CUSTOMER_NUMBER
- CUSTOMER_NAME
- [MORE…..]

MONTHLY_ACCOUNT
- MONTH_END_DATE_ID
- ACCOUNT_ID
- CUSTOMER_ID
- [PREVIOUS BALANCE]
- [TOTAL DEPOSITS]
- [TOTAL WITHDRAWL]
- [AVAILABLE BALANCE]

MONTH
- MONTH_END_DATE_ID
- [MONTH NAME]
- [CALENDAR MONTH]
- [MORE…]

1 Row for every Customer every Month

Steps to Create Dimensional Modelling

- Identify Grain
  - Describes the level of detail for the business problem/solution
  - The lowest level of information for any table in your data warehouse
  - Need to answer questions:
    - Do we need to store all the available products or just a few types of products?
    - Do we store the product sale information on a monthly, weekly, daily or hourly basis?
    - How do the above two choices affect the database size?

Steps to Create Dimensional Modelling

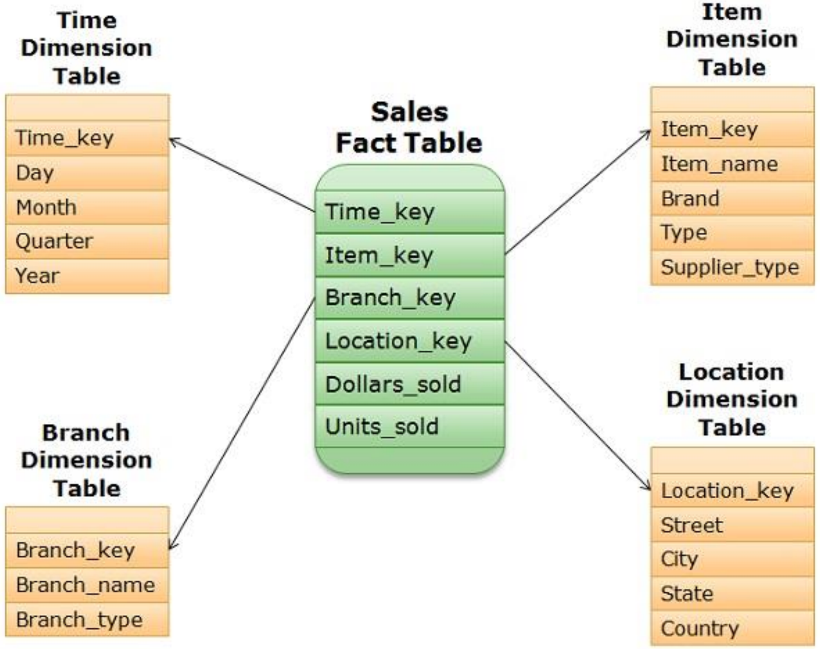
- Identify Dimensions
  - Dimensions provide context for facts
  - We can easily identify dimensions because of “by”, and/or “for” words. Like date, store, inventory, etc.
    - Ex. we want to find the sales for specific products in different locations on a daily basis.
      - Dimensions: Product, Location and Time
      - Attributes: for Product: Product key (Foreign Key), Name, Type, Specifications
      - Hierarchies: For Location: Country, State, City, Street Address, Name

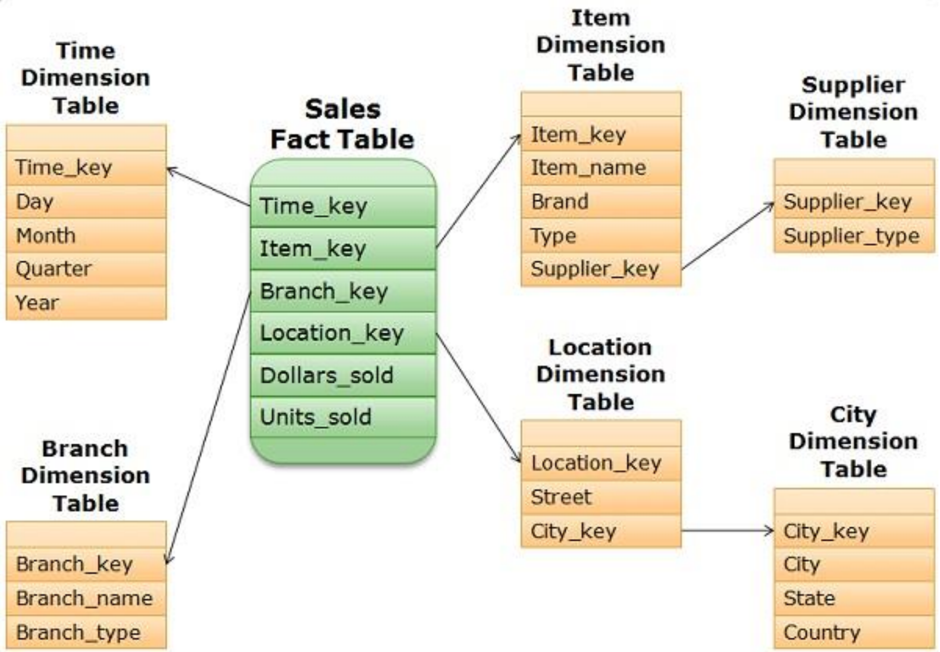
Steps to Create Dimensional Modelling

- Identify Facts
  - Facts are quantifiable numerical values associated with the business process
  - This step is co-associated with the business users (they get access to data).
  - Most of the fact table rows are numerical values like price or cost per unit, etc.
  - Ex. we want to find the sales for specific products in different locations on a daily basis.
    - The fact is Sum of Sales by product by location by time.

Steps to Create Dimensional Modelling

- Build Schema
  - Star Schema
  - Snowflake Schema





Rules for Dimensional Modelling

- Load atomic (nguyên tử) data into dimensional structures.
- Build dimensional models around business processes.
- Every fact table has an associated date dimension table.
- All facts in a single fact table are at the same grain or level of detail.
- Dimension tables use a surrogate key (không liên hệ với DL, ex. auto number)
- Balance requirements and realities to deliver business solution to support their decision-making

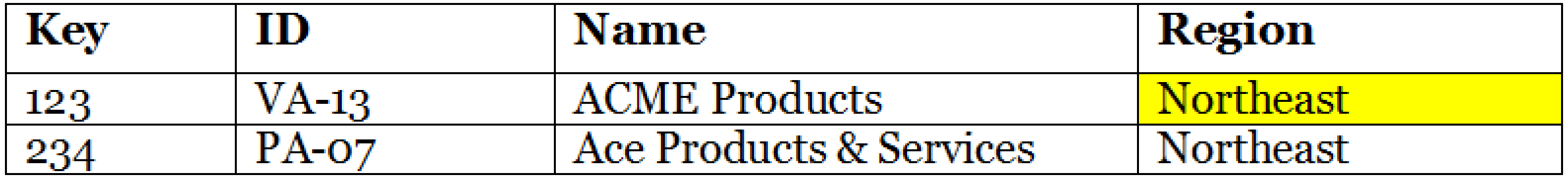
Slowly Changing Dimensions

- Dimensional data changes infrequently but when it does, you need a strategy for addressing the change.
  - Ex. What happens when a customer has a new address, an employee has a name change?
- **4 popular Strategies**
  - Type 1: Overwrite the existing attribute
  - Type 2: Add a new Dimension row
  - Type 3: Add a new dimension attribute
  - Type 6: (1+2+3)

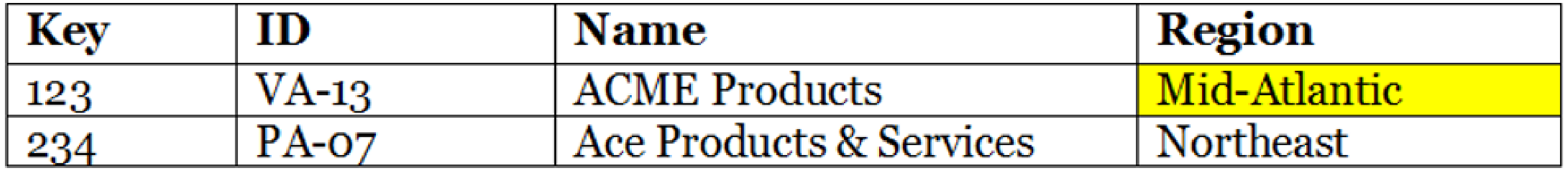
Slowly Changing Dimensions

Type 1: Overwrite the existing attribute

- Appropriate for
  - Correcting mistakes or error in data
  - Change where historical associations do not matter
  - The old value has no significance
- Ex. Employee name changes, Corrections…



Key	ID	Name	Region
123	VA-13	ACME Products	Northeast
234	PA-07	Ace Products & Services	Northeast

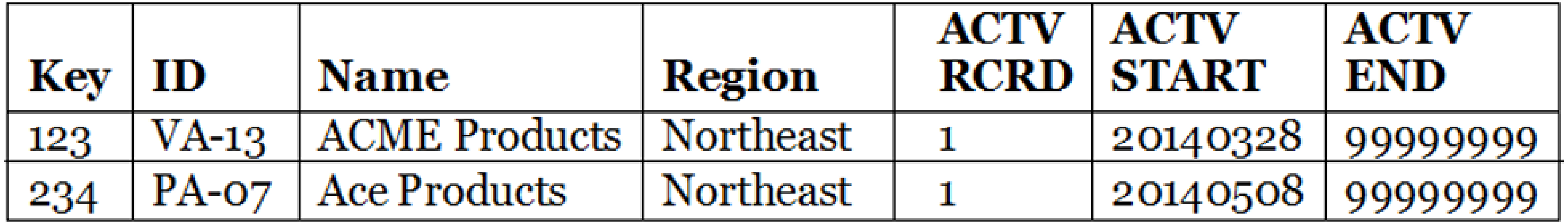


Key	ID	Name	Region
123	VA-13	ACME Products	Mid-Atlantic
234	PA-07	Ace Products & Services	Northeast

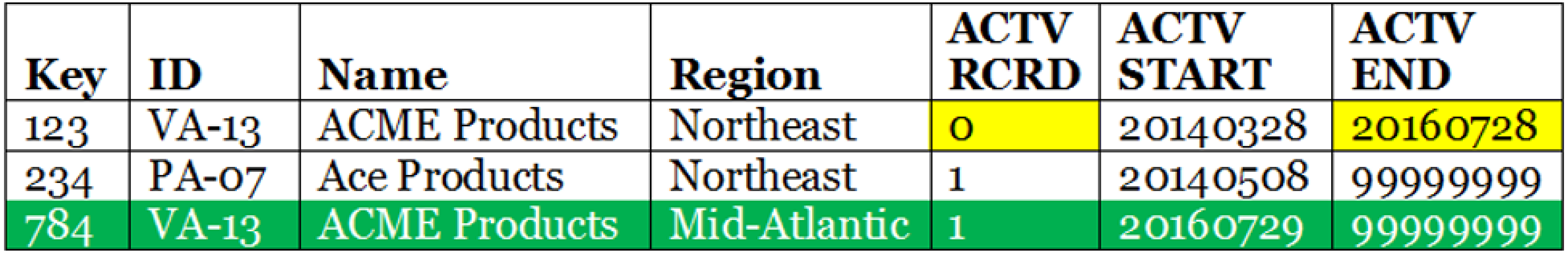
Slowly Changing Dimensions

Type 2: Add a new Dimension row

- Most popular strategy, as it preserves history
- Natural key is repeated (surrogate key (không liên hệ với DL, ex. auto number)
- Old and new values are stored along with effective dates and indicator of which row is "current"



Key	ID	Name	Region	ACTV RCRD	ACTV START	ACTV END
123	VA-13	ACME Products	Northeast	1	20140328	99999999
234	PA-07	Ace Products	Northeast	1	20140508	99999999

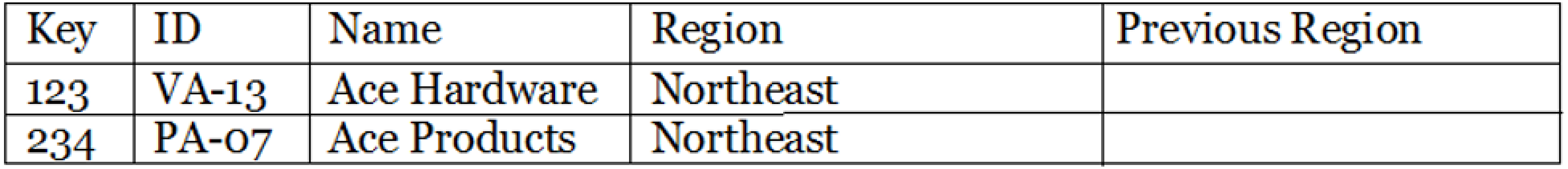


Key	ID	Name	Region	ACTV RCRD	ACTV START	ACTV END
123	VA-13	ACME Products	Northeast	0	20140328	20160728
234	PA-07	Ace Products	Northeast	1	20140508	99999999
784	VA-13	ACME Products	Mid-Atlantic	1	20160729	99999999

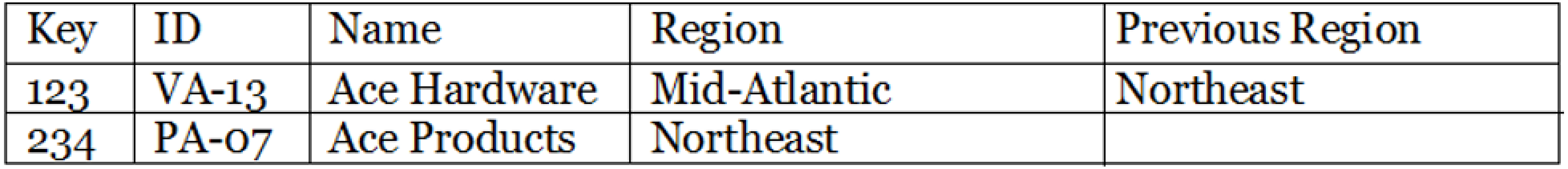
Slowly Changing Dimensions

Type 3: Add a new dimension attribute

- Infrequently used, preserves history
- Useful for soft changes where users might want to choose between the old and new attribute, or need to access both values for a time
- The new value is written to the existing column, the old value is stored in a new column
- This way queries do not have to be re-written to access the new attribute



Key	ID	Name	Region	Previous Region
123	VA-13	Ace Hardware	Northeast	
234	PA-07	Ace Products	Northeast	

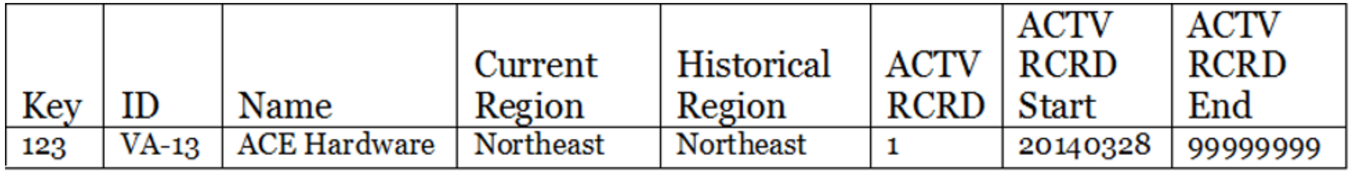


Key	ID	Name	Region	Previous Region
123	VA-13	Ace Hardware	Mid-Atlantic	Northeast
234	PA-07	Ace Products	Northeast	

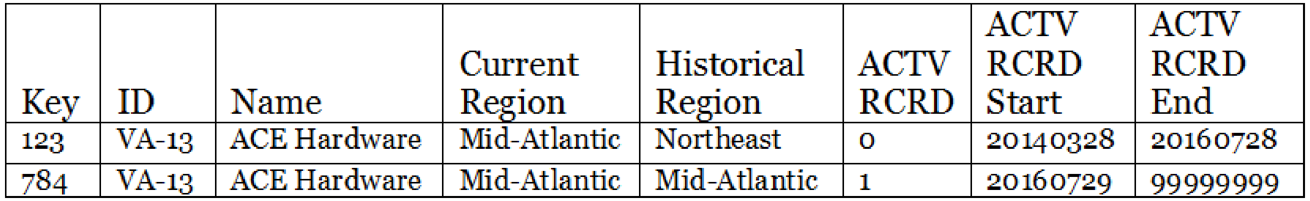
Slowly Changing Dimensions

Type 6: (1+2+3)

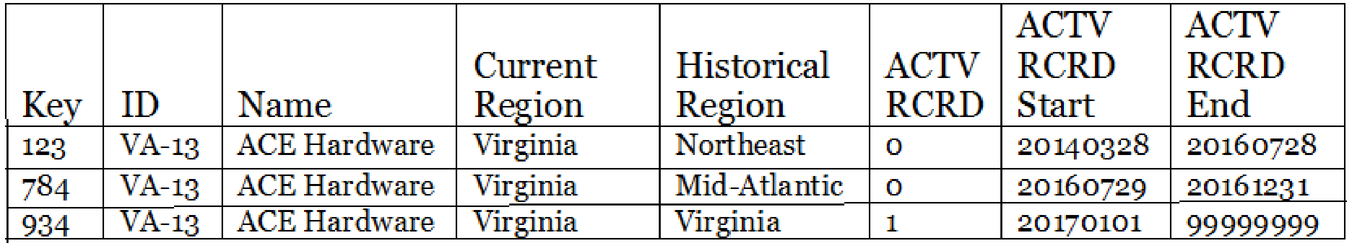
- Type 6 is a very rarely used (hiếm khi sử dụng)
- Start with a Type 2, add columns for the records you wish to capture the current value as well as the historical value. This allows one to filter or group on the Type 2 value in effect when the measure occurred or the current attribute value



Key	ID	Name	Current Region	Historical Region	ACTV RCRD	ACTV RCRD Start	ACTV RCRD End
123	VA-13	ACE Hardware	Northeast	Northeast	1	20140328	99999999



Key	ID	Name	Current Region	Historical Region	ACTV RCRD	ACTV RCRD Start	ACTV RCRD End
123	VA-13	ACE Hardware	Mid-Atlantic	Northeast	0	20140328	20160728
784	VA-13	ACE Hardware	Mid-Atlantic	Mid-Atlantic	1	20160729	99999999



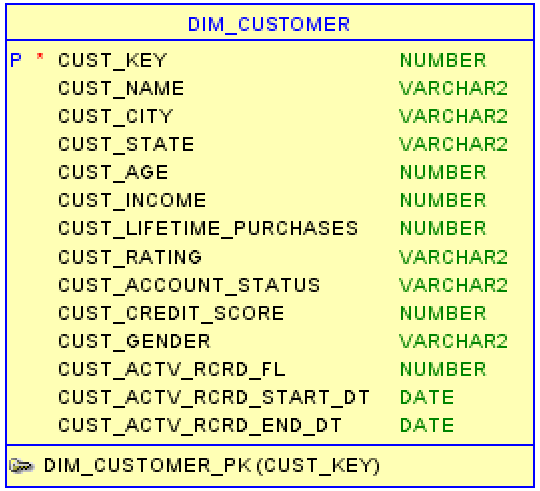
Key	ID	Name	Current Region	Historical Region	ACTV RCRD	ACTV RCRD Start	ACTV RCRD End
123	VA-13	ACE Hardware	Virginia	Northeast	0	20140328	20160728
784	VA-13	ACE Hardware	Virginia	Mid-Atlantic	0	20160729	20161231
934	VA-13	ACE Hardware	Virginia	Virginia	1	20170101	99999999

Rapidly Changing Dimensions

- The attribute values of dimension change frequently causing the dimension grow rapidly
- The rapid growth of this dimension will impact maintenance and performance as the dimension grows.

Solution: Mini-Dimension

- Mini-dimensions contain the rapidly changing attributes of the original dimension

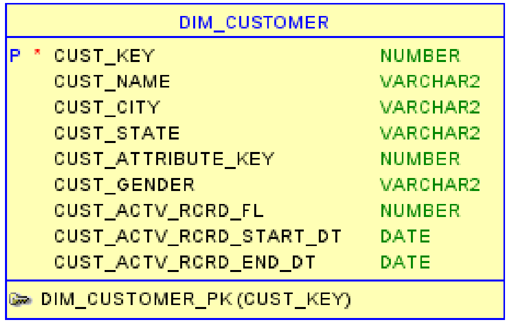


DIM_CUSTOMER

P *	CUST_KEY	NUMBER
	CUST_NAME	VARCHAR2
	CUST_CITY	VARCHAR2
	CUST_STATE	VARCHAR2
	CUST_AGE	NUMBER
	CUST_INCOME	NUMBER
	CUST_LIFETIME_PURCHASES	NUMBER
	CUST_RATING	VARCHAR2
	CUST_ACCOUNT_STATUS	VARCHAR2
	CUST_CREDIT_SCORE	NUMBER
	CUST_GENDER	VARCHAR2
	CUST_ACTV_RCRD_FL	NUMBER
	CUST_ACTV_RCRD_START_DT	DATE
	CUST_ACTV_RCRD_END_DT	DATE

DIM_CUSTOMER_PK (CUST_KEY)

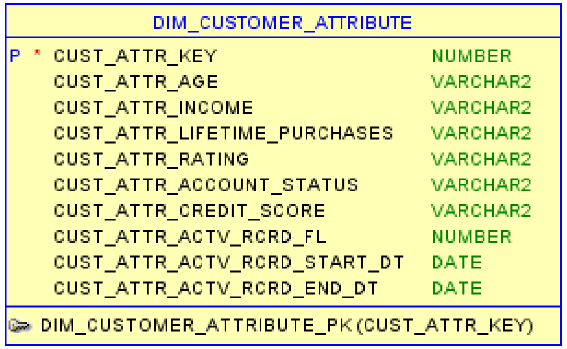
Rapidly changing attributes: customer's age, income, the number of lifetime purchases, rating, account status, and credit score



DIM_CUSTOMER

P *	CUST_KEY	NUMBER
	CUST_NAME	VARCHAR2
	CUST_CITY	VARCHAR2
	CUST_STATE	VARCHAR2
	CUST_ATTRIBUTE_KEY	NUMBER
	CUST_GENDER	VARCHAR2
	CUST_ACTV_RCRD_FL	NUMBER
	CUST_ACTV_RCRD_START_DT	DATE
	CUST_ACTV_RCRD_END_DT	DATE

DIM_CUSTOMER_PK (CUST_KEY)



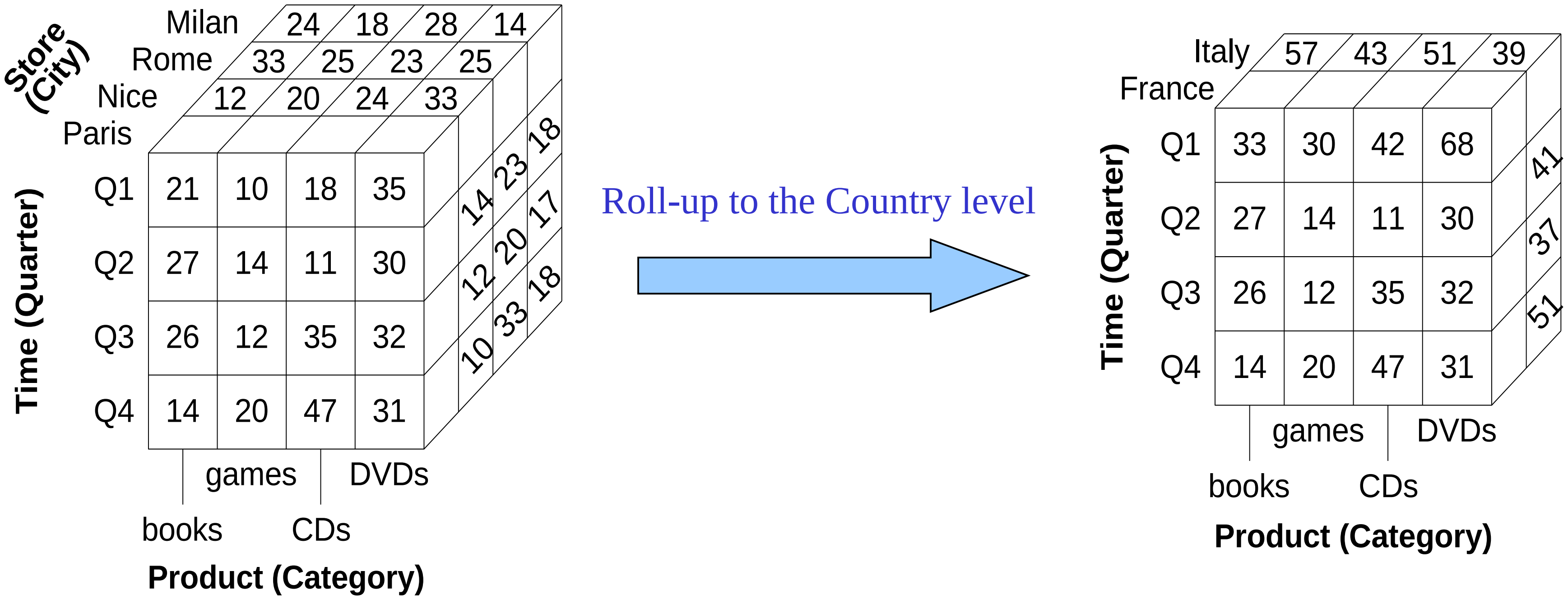
DIM_CUSTOMER_ATTRIBUTE

P *	CUST_ATTR_KEY	NUMBER
	CUST_ATTR_AGE	VARCHAR2
	CUST_ATTR_INCOME	VARCHAR2
	CUST_ATTR_LIFETIME_PURCHASES	VARCHAR2
	CUST_ATTR_RATING	VARCHAR2
	CUST_ATTR_ACCOUNT_STATUS	VARCHAR2
	CUST_ATTR_CREDIT_SCORE	VARCHAR2
	CUST_ATTR_ACTV_RCRD_FL	NUMBER
	CUST_ATTR_ACTV_RCRD_START_DT	DATE
	CUST_ATTR_ACTV_RCRD_END_DT	DATE

DIM_CUSTOMER_ATTRIBUTE_PK (CUST_ATTR_KEY)

OLAP Operations: Roll up

- Transforms detailed measures into summarized ones when one moves up in a hierarchy



Store (City)

Time (Quarter)

	books	games	CDs	DVDs
Milan	24	18	28	14
Rome	33	25	23	25
Nice	12	20	24	33
Paris				

Paris	books	games	CDs	DVDs
Q1	21	10	18	35
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

DVDs	Nice	Rome	Milan
Q1	14	23	18
Q2	12	20	17
Q3	10	33	18

Product (Category)

Roll-up to the Country level

Time (Quarter)

	books	games	CDs	DVDs
Italy	57	43	51	39
France				

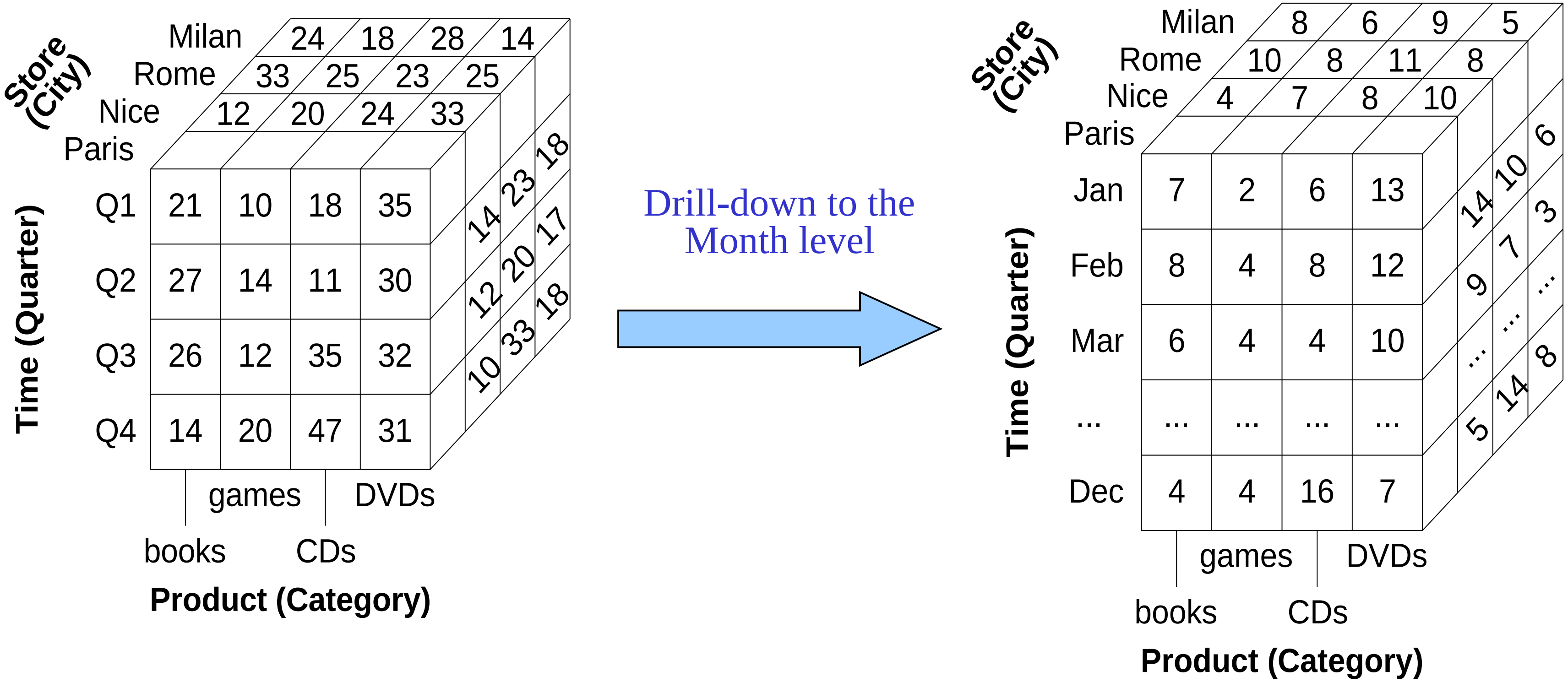
France	books	games	CDs	DVDs
Q1	33	30	42	68
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

DVDs	Italy
Q1	41
Q2	37
Q3	51

Product (Category)

OLAP Operations: Drill down

- Opposite to the roll-up operation, i.e., it moves from a more general level to a detailed level in a hierarchy



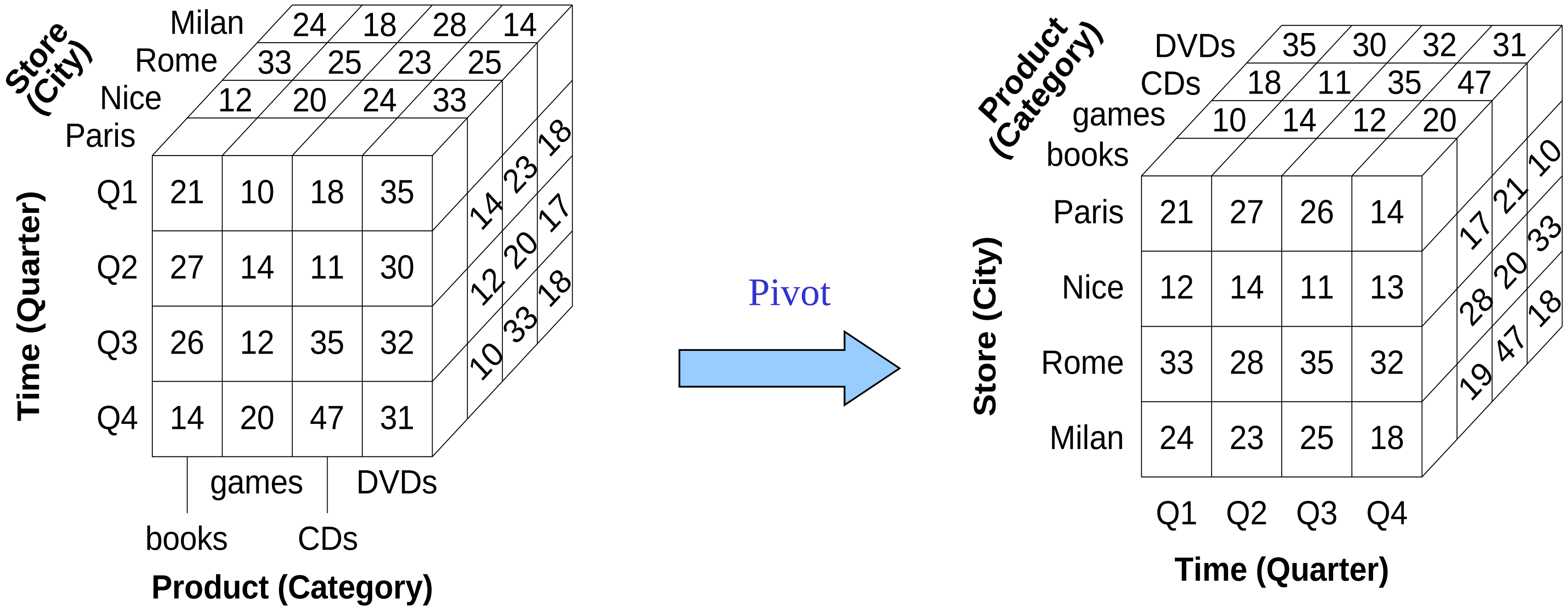
Drill-down to the Month level

Paris	books	games	CDs	DVDs
Q1	21	10	18	35
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

Paris	books	games	CDs	DVDs
Jan	7	2	6	13
Feb	8	4	8	12
Mar	6	4	4	10
...	...	...	...	...
Dec	4	4	16	7

OLAP Operations: Pivot or Rotate

- Rotates the axes of a cube to provide an alternative presentation of the data



Time (Quarter)	books	games	CDs	DVDs
Q1	21	10	18	35
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

Milan 24 18 28 14

Rome 33 25 23 25

Nice 12 20 24 33

14 23 18

12 20 17

10 33 18

Pivot

Store (City)	Q1	Q2	Q3	Q4
Paris	21	27	26	14
Nice	12	14	11	13
Rome	33	28	35	32
Milan	24	23	25	18

DVDs 35 30 32 31

CDs 18 11 35 47

games 10 14 12 20

books

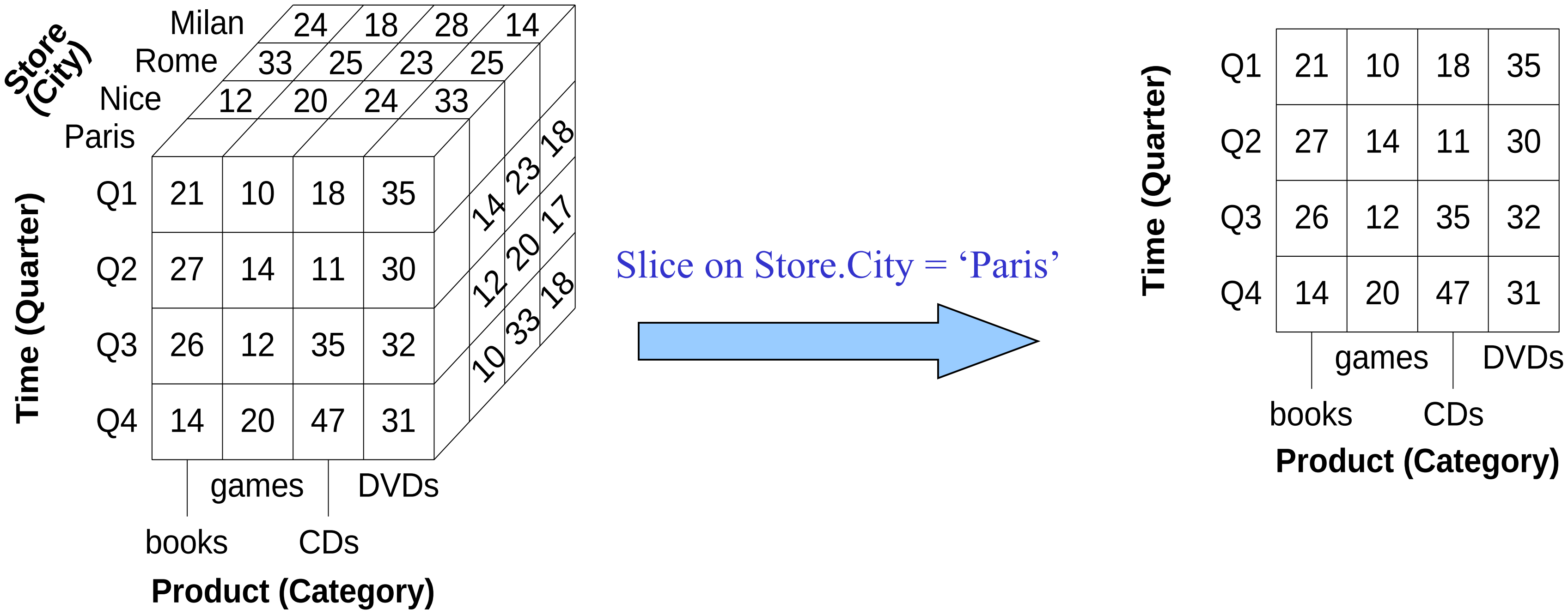
17 21 10

28 20 33

19 47 18

OLAP Operations: Slice

- Performs a selection on one dimension of a cube, resulting in a subcube



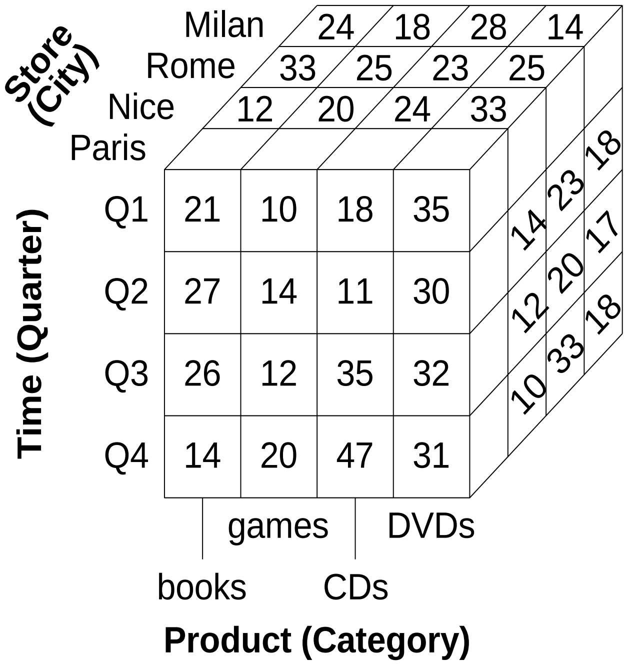
Slice on Store.City = 'Paris'

Time (Quarter)	books	games	CDs	DVDs
Q1	21	10	18	35
Q2	27	14	11	30
Q3	26	12	35	32
Q4	14	20	47	31

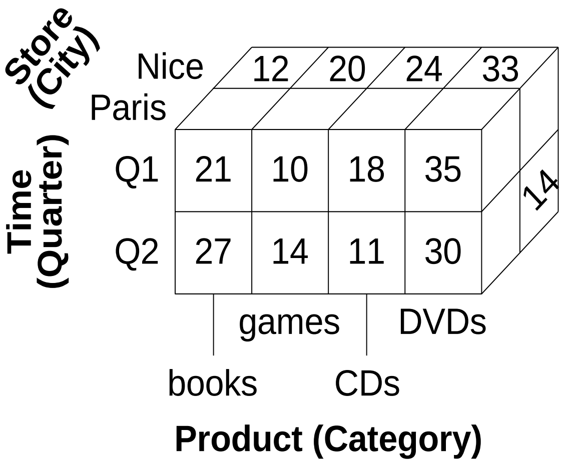
Product (Category)

OLAP Operations: Dice

- Defines a selection on two or more dimensions, thus again defining a subcube



Dice on Store.Country = ‘France’
and Time.Quarter= ‘Q1’ or ‘Q2’

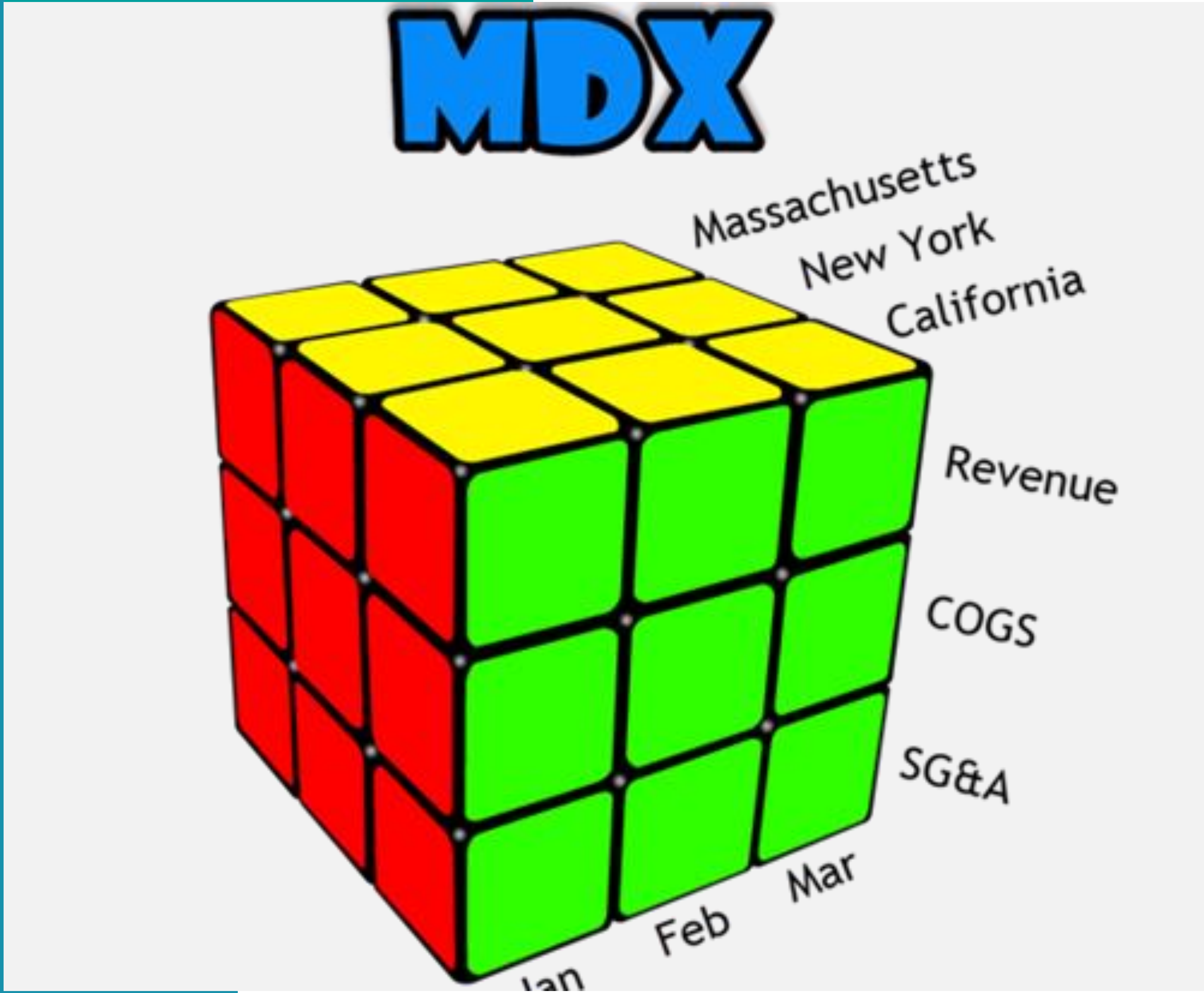


Questions

1. How to build dimensional modelling?
2. What is Slowly changing dimension?
3. What is Rapidly changing dimension?
4. What is OLAP (Online Analytical Processing)?
5. What are OLAP operations?

DATA WAREHOUSE

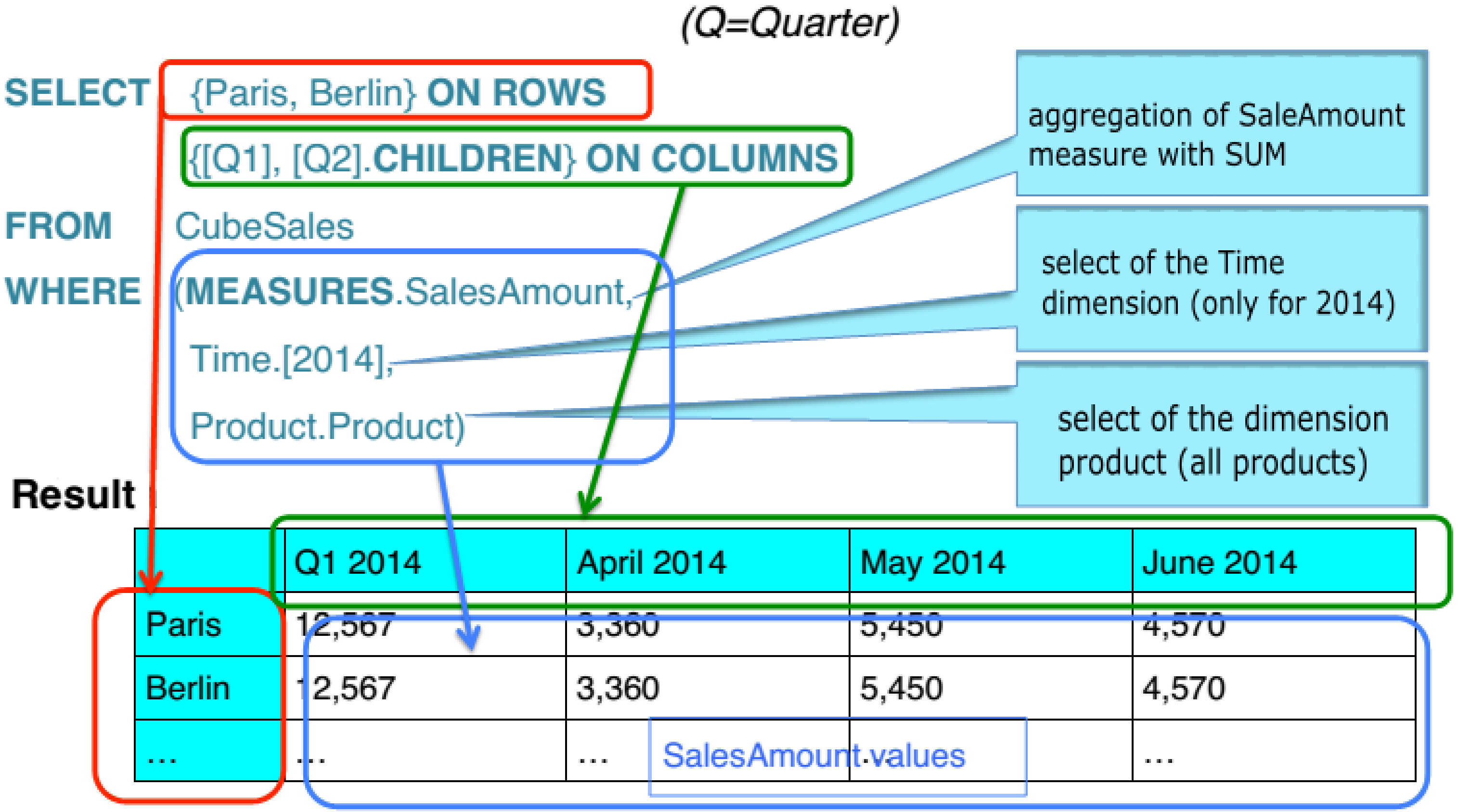
- MDX Definition
- MDX Query Syntax
- Demo MDX



MDX (MultiDimensional Expressions)

- Syntax (basic)
  - SELECT <measures|dimensions> [on columns|rows]
  - FROM <Cube>
  - [WHERE <Slicing conditions>]
- How to use MDX?
- Create Datawarehouse (OLAP|Cube) using SSIS
- Using MDX to query Data in the cube

MDX (MultiDimensional Expressions)



(Q=Quarter)

```
SELECT   {Paris, Berlin} ON ROWS
         {[Q1], [Q2].CHILDREN} ON COLUMNS
FROM     CubeSales
WHERE    (MEASURES.SalesAmount,
          Time.[2014],
          Product.Product)
```

aggregation of SaleAmount measure with SUM

select of the Time dimension (only for 2014)

select of the dimension product (all products)

Result

	Q1 2014	April 2014	May 2014	June 2014
Paris	12,567	3,360	5,450	4,570
Berlin	12,567	3,360	5,450	4,570
…	…	…		…

MDX (MultiDimensional Expressions)

- **Demo**

  1. Total Sales

  2. Total Sales by Country

  2a. Total Sales of Vietnam

  2b. Total Sales of Vietnam

  3. Total Sales by Year

  4. Total Sales by Year (skip null value)

  5. Total Sales by Year of Vietnam

  6. Total Sales and Profit by Year (skip null value)

  7. Total Sale by Year and Sub-Category

DATA WAREHOUSE

- What is Meta Data?
- Why metadata is important
- A Critical Need in the Data Warehouse
- RAID Technology

What is Meta Data?

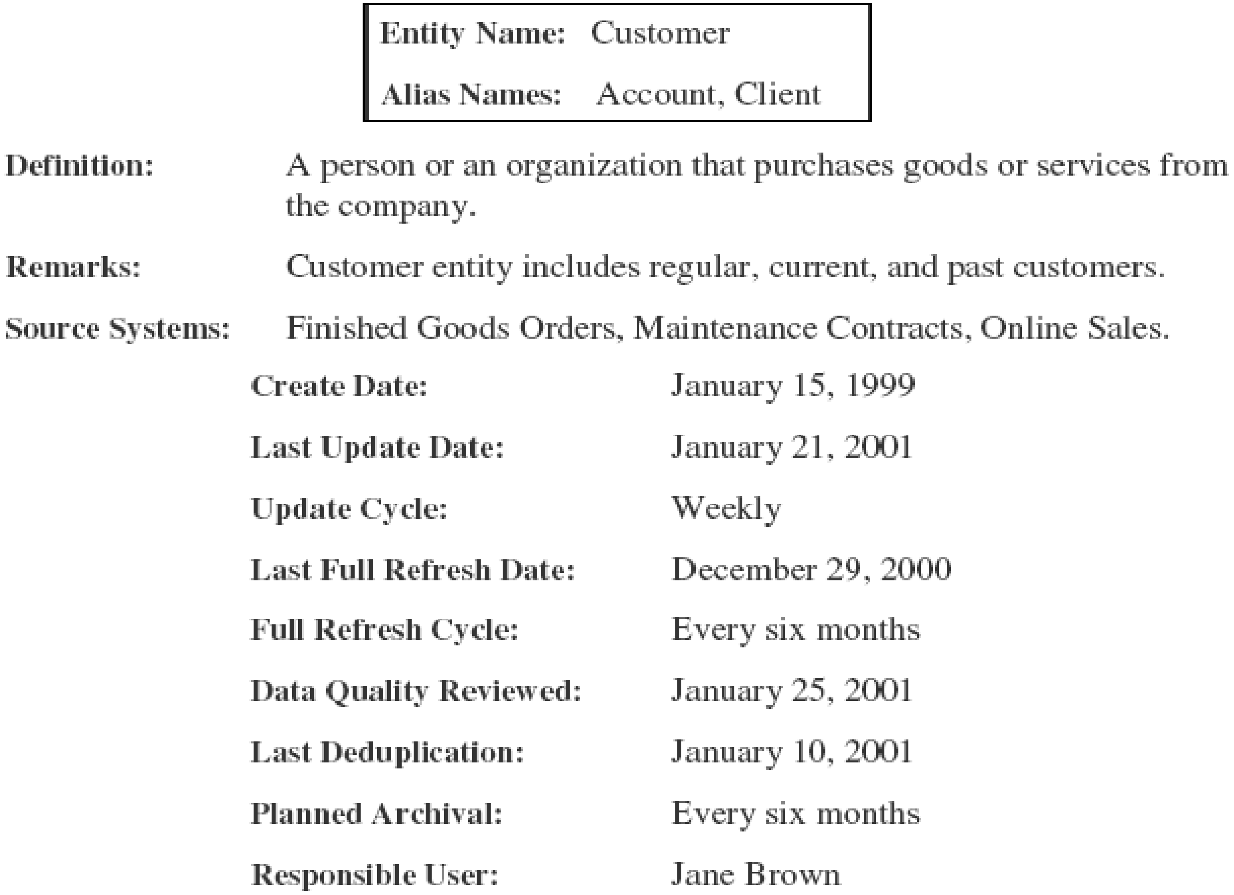
- Metadata is data about the data or documentation about the information which is required by the users.
- Metadata includes the following:
  - The location and descriptions of warehouse systems and components.
  - Names, definitions, structures, and content of data-warehouse and end-users views.
  - Identification of authoritative data sources.
  - Integration and transformation rules used to populate data.
  - Integration and transformation rules used to deliver information to end-user analytical tools.
  - Subscription information for information delivery to analysis subscribers.
  - Metrics used to analyze warehouses usage and performance.
  - Security authorizations, access control list, etc.

Why metadata is important

- Metadata in a data warehouse contains the answers to questions about the data in the data warehouse.
  - You keep the answers in a place called the metadata repository.
  - Here is a sample list of definitions:
    - Data about the data
    - Table of contents for the data
    - Catalog for the data
    - Data warehouse atlas (bản đồ)
    - Data warehouse roadmap (lộ trình)
    - Data warehouse directory
    - Glue that holds the data warehouse contents together

Why metadata is important

- Metadata element for the Customer entity



Entity Name: Customer

Alias Names: Account, Client

Definition: A person or an organization that purchases goods or services from the company.

Remarks: Customer entity includes regular, current, and past customers.

Source Systems: Finished Goods Orders, Maintenance Contracts, Online Sales.

Create Date:	January 15, 1999
Last Update Date:	January 21, 2001
Update Cycle:	Weekly
Last Full Refresh Date:	December 29, 2000
Full Refresh Cycle:	Every six months
Data Quality Reviewed:	January 25, 2001
Last Deduplication:	January 10, 2001
Planned Archival:	Every six months
Responsible User:	Jane Brown

Figure 9-1 Metadata element for *Customer* entity.

A Critical Need in the Data Warehouse

- For Using the Data Warehouse
  - Users retrieve information from the data warehouse.
  - Users themselves create ad hoc queries and run these against the data warehouse.
  - They format their own reports.
  - Before they can create and run their queries, users need to know about the data in the data warehouse

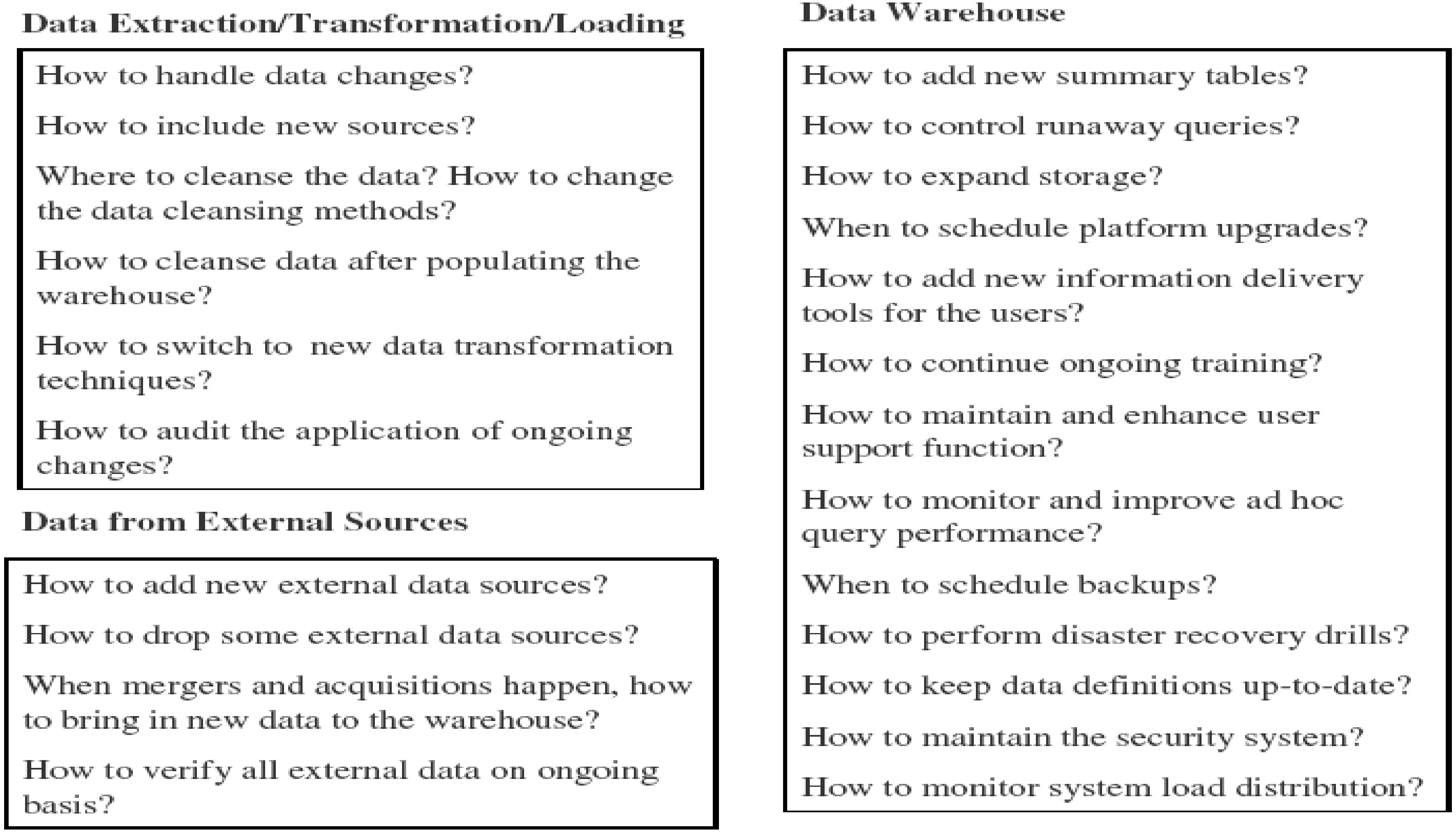
    => They need metadata.

A Critical Need in the Data Warehouse

- For Building the Data Warehouse.
  - Metadata is absolutely essential for building your data warehouse in every activity and every task.
    - Know the source systems and their data structures.
    - Know the structures and the data content in the data warehouse.
    - Determine the mappings and the data transformations.

A Critical Need in the Data Warehouse

- For Administering the Data Warehouse



Data Extraction/Transformation/Loading

How to handle data changes?

How to include new sources?

Where to cleanse the data? How to change the data cleansing methods?

How to cleanse data after populating the warehouse?

How to switch to new data transformation techniques?

How to audit the application of ongoing changes?

Data from External Sources

How to add new external data sources?

How to drop some external data sources?

When mergers and acquisitions happen, how to bring in new data to the warehouse?

How to verify all external data on ongoing basis?

Data Warehouse

How to add new summary tables?

How to control runaway queries?

How to expand storage?

When to schedule platform upgrades?

How to add new information delivery tools for the users?

How to continue ongoing training?

How to maintain and enhance user support function?

How to monitor and improve ad hoc query performance?

When to schedule backups?

How to perform disaster recovery drills?

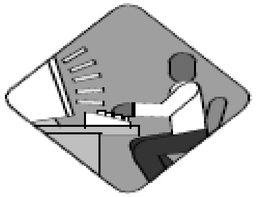
How to keep data definitions up-to-date?

How to maintain the security system?

How to monitor system load distribution?

A Critical Need in the Data Warehouse

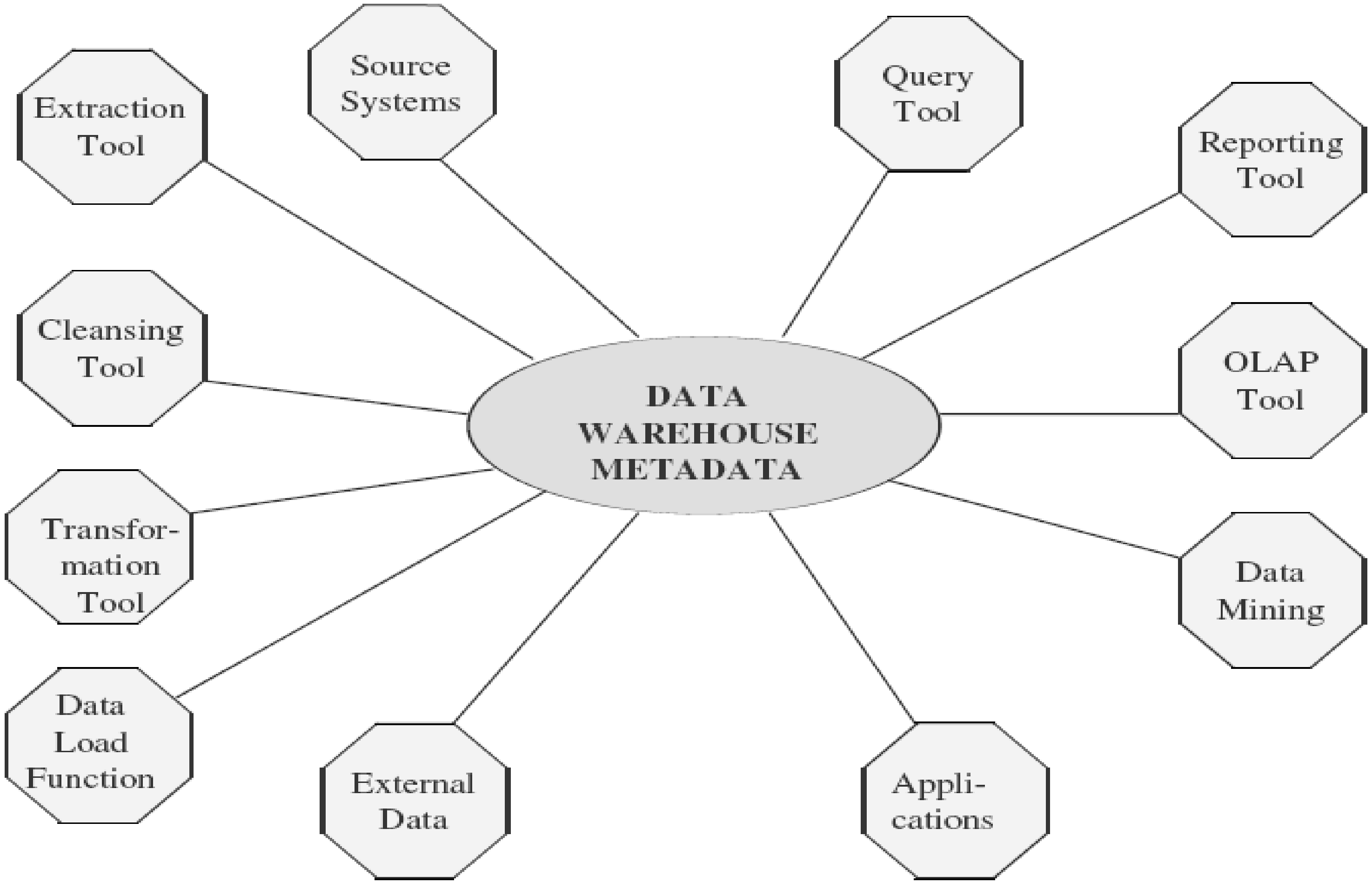
Who Needs Metadata



	IT Professionals	Power Users	Casual Users
Information Discovery	Databases, Tables, Columns, Server Platforms	Databases, Tables, Columns	List of Predefined Queries and Reports, Business Views
Meaning of Data	Data Structures, Data Definitions, Data Mapping, Cleansing Functions, Transformation Rules	Business Terms, Data Definitions, Data Mapping, Cleansing Functions, Transformation Rules	Business Terms, Data Definitions, Filters, Data Sources, Conversion , Data Owners
Information Access	Program Code in SQL, 3GL,4GL, Front-end Applications, Security	Query Toolsets, Database Access for Complex Analysis	Authorization Requests, Information Retrieval into Desktop Applications such as Spreadsheets

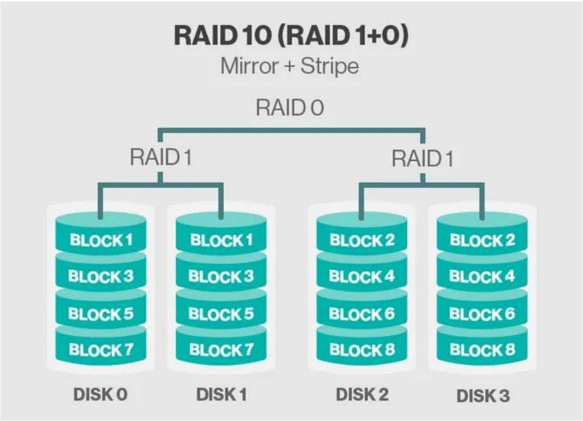
A Critical Need in the Data Warehouse

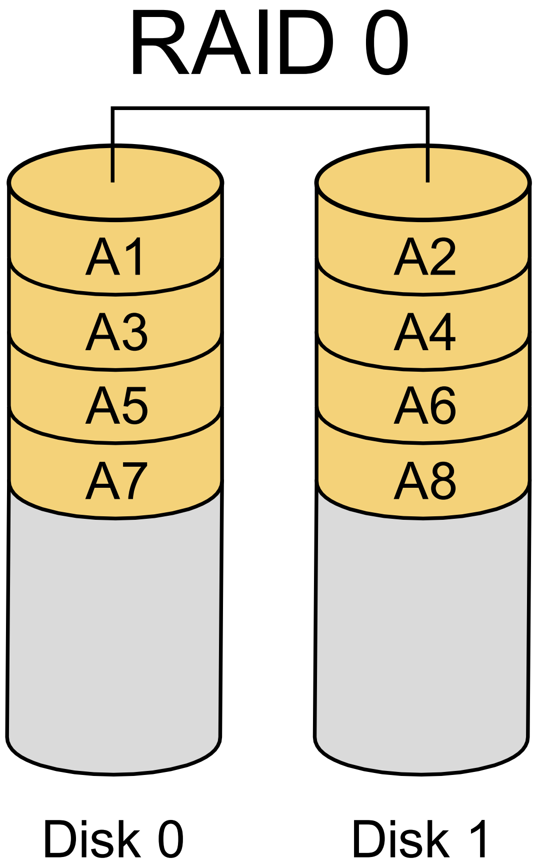
- Metadata is Like a Nerve Center

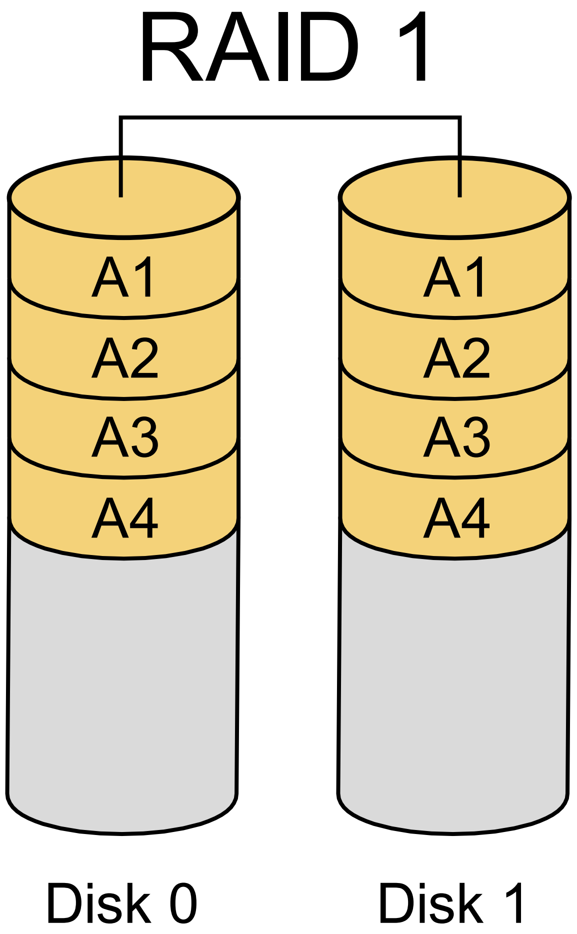


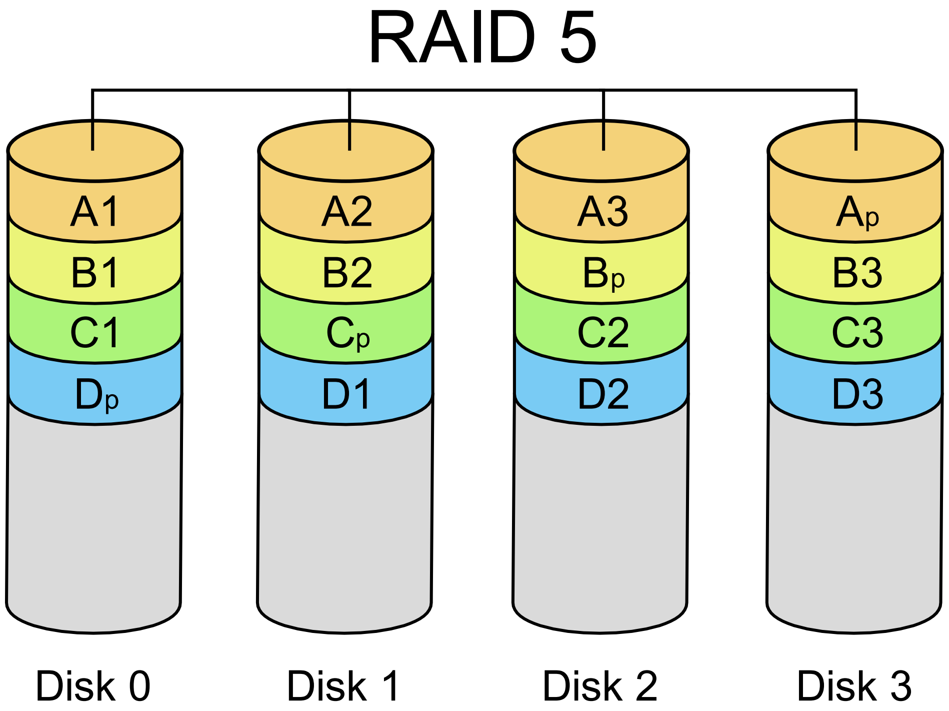
RAID Technology

- Some types of RAID









Questions

1. What is MDX (MultiDimensional eXpressions) language? How do we use MDX to retrieve data in a data warehouse?
2. What is metadata? What are metadata types? (methods for classification of metadata; detail in Metadata types by functional areas)
3. What is the RAID technology? Explain more detail RAID 0, 1, 5.