MIS781 Business Intelligence and Database

Module 6: Data Warehouse

Jokes ☺

1. Why is the Australian emergency line is“000”?

Because it’ll look the same when your phone is upside down ☺

2. Why do kangaroo mums hate bad weather?

Their joeys have to play inside ☺

3. Why did the cockatoo sit on the clock?

So he would be on time ☺

Learning Objectives

By the end of this class, you should be able to:

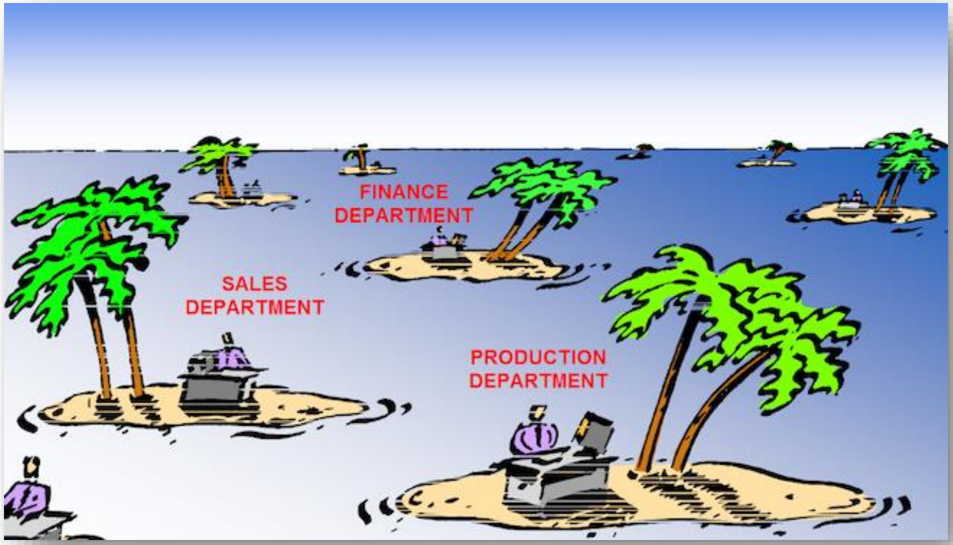
- Articulate the differences between **transactional** and **analytical** (informational) databases
- Explain and exemplify the **characteristics of a Data Warehouse**
- Explain **Dimensional Model** & **schema**
- Explain **Online Analytical Processing (OLAP) and Multi-dimensionality Concept**

Crisis of the Classical Information System

Problem

- Despite huge quantities of data availability, why can't managers use it for strategic decision making?
- Data is useful for business operations but not for decision making.

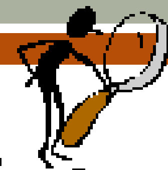
Why so ?



- Data of an enterprise is spread across many types of incompatible structures and systems, multiple platforms ('Info Islands').
- Data for strategic decision making not in suitable format for easy analysis.

We are not sure because:

Crisis of the Classical Information System


Nobody has the information

The information was not collected

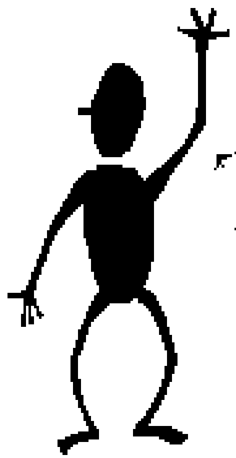
Nobody thought to collect it
Nobody knew how to collect it
Nobody asked for it to be collected
Somebody prevented its collection
Taboos forbid its collection or hid its existence

The information was too expensive to collect

The information was too complex to collect
The information was too scattered to collect

The question was wrong

Alternative information might answer the need
The information was impossible to collect
Information was collected but was not what was needed





Somebody does have the information but:

They will not reveal the information

The information is held secret
The holder of the information wants too much money for it
The holder of the information is too shy to reveal it
The holder of the information is afraid to reveal it

Nobody has asked for the information

Nobody realised the importance of the information
The information was too obvious to seem important
The information was harmful to the ones who might ask for it

The information is too complex to understand

There is too much information
The information is fragmented and not comparable
The information is in an inaccessible form
(language or technology)

The information is not believed

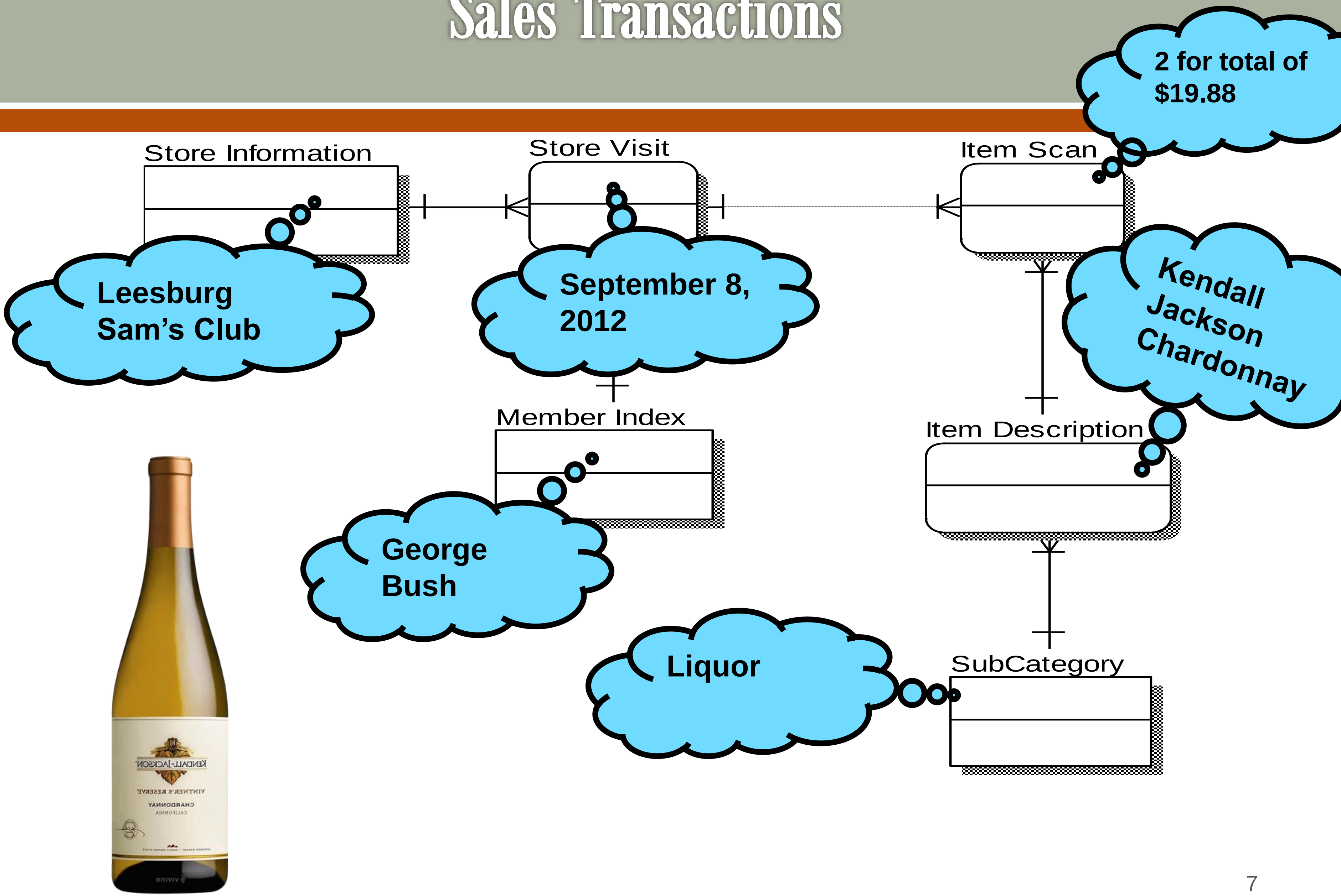
There is a competing set of information or belief
The question is wrong but gives an acceptable answer
so other data are ignored.

Vignette: a wine export company

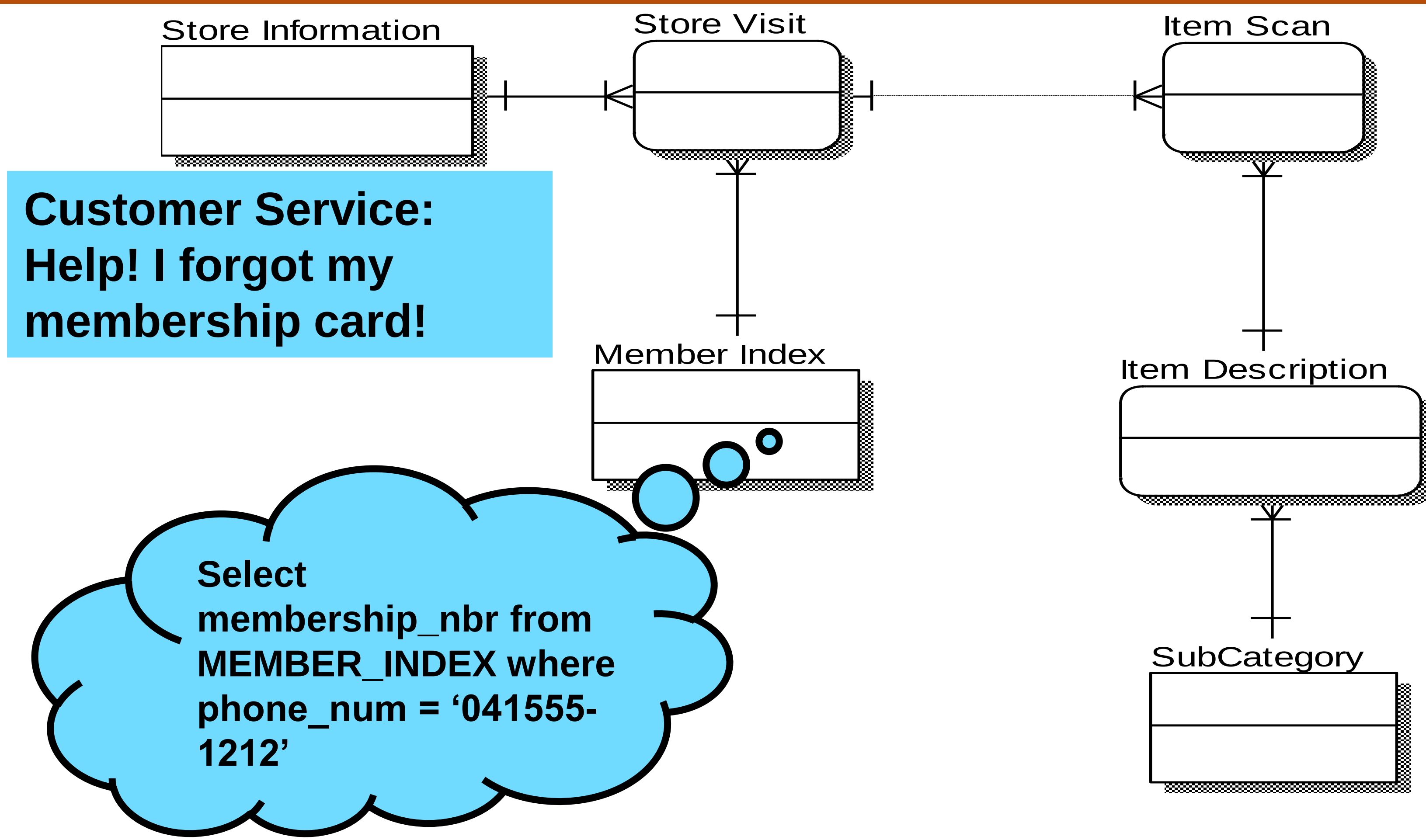
- Consider some strategic questions the directors of a 'wine export company' may ask,
  - Which product lines are increasing (or decreasing) in popularity over time?
  - Which customers place the same orders on a regular basis?
  - Are some products more popular in different markets?
  - Is there a demand for wine from a specific region in a specific market during a specific period? ('geo-seasonality')




Sales Transactions

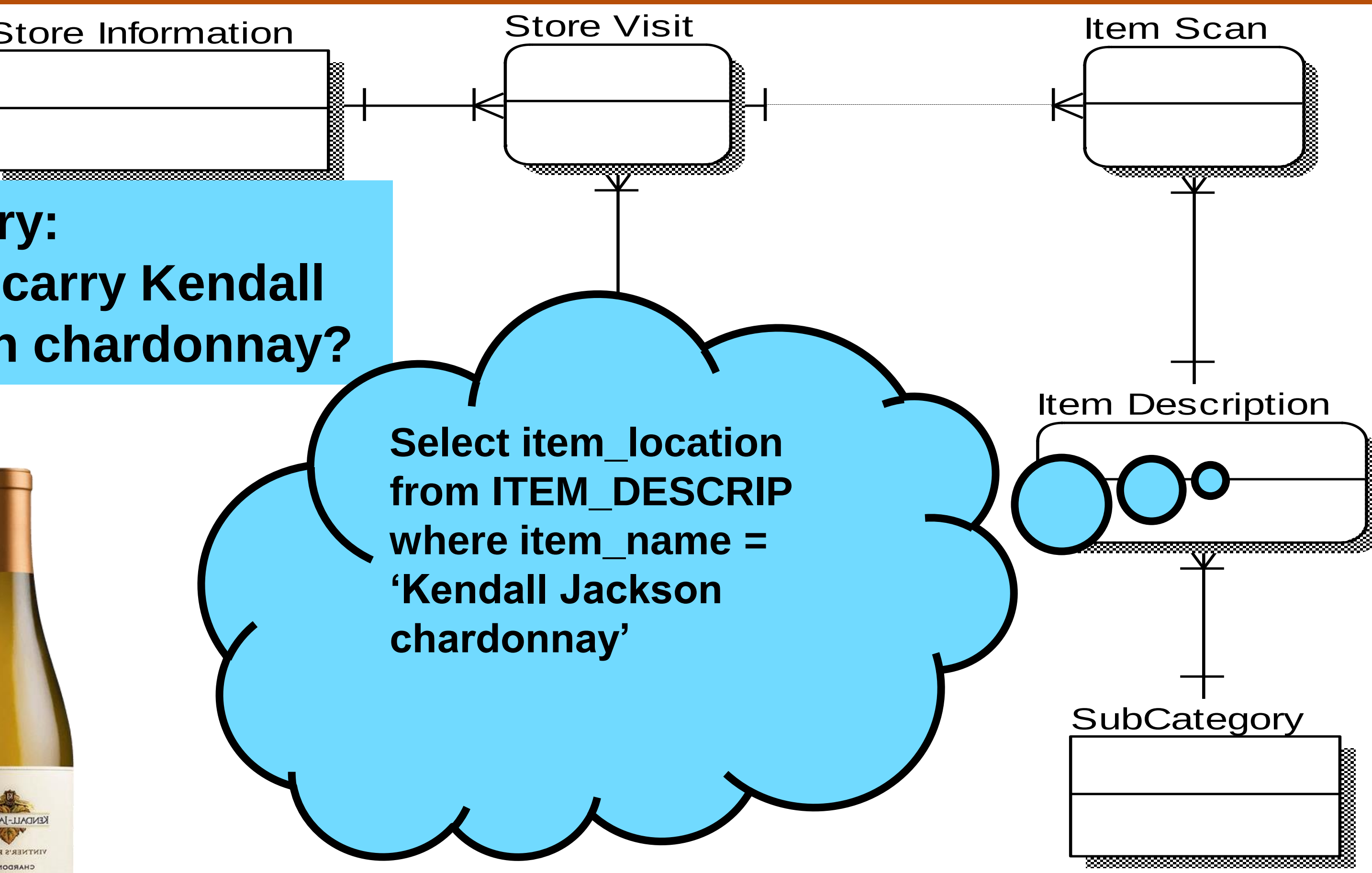


Transactional (Operational) Questions



Transactional (Operational)

Inventory:
Do you carry Kendall Jackson chardonnay?

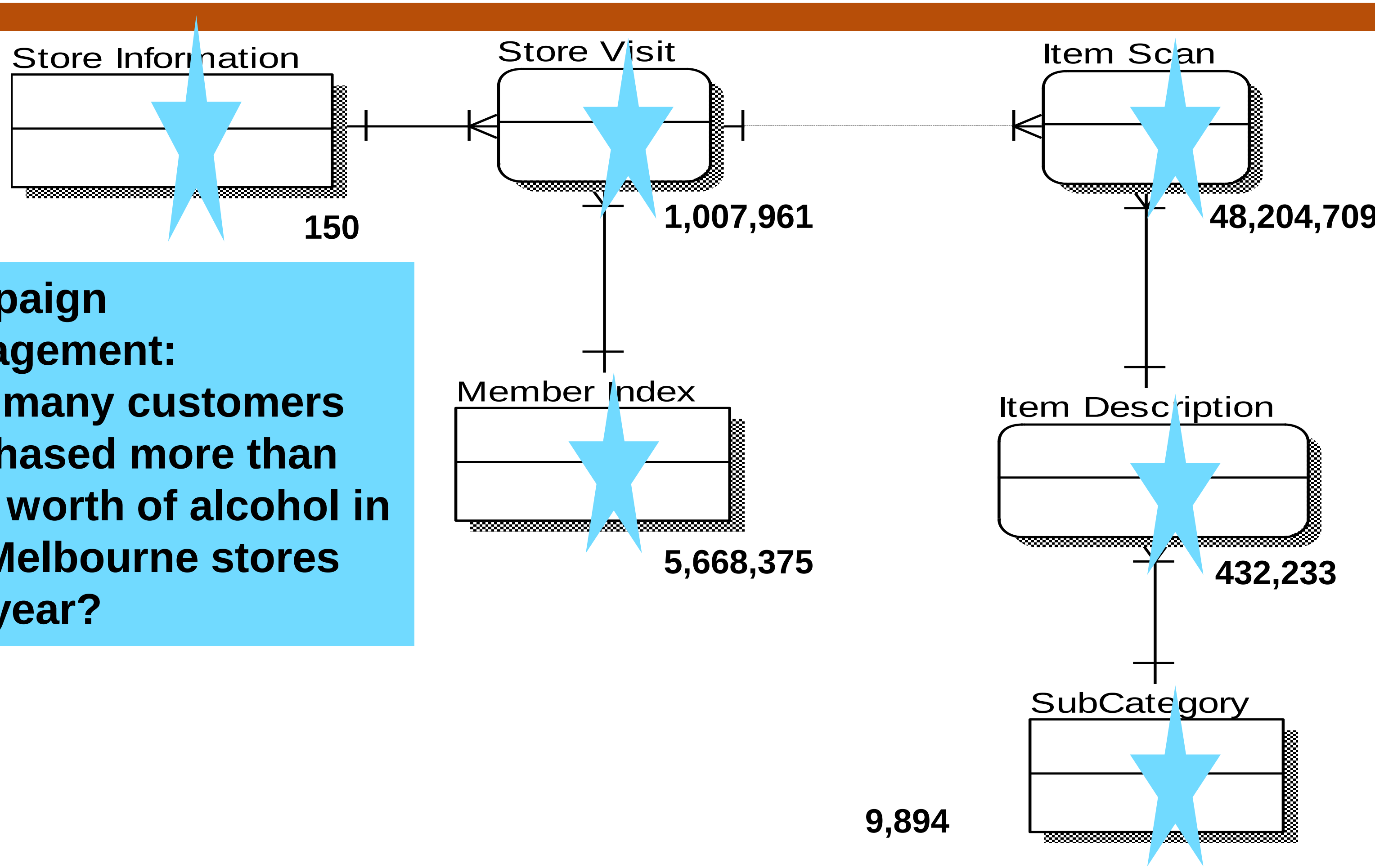


Select item_location from ITEM_DESCRIP where item_name = 'Kendall Jackson chardonnay'

Transactional (Operational)

Customer Service: What stores are open on Sunday in Melbourne?

Analytical Questions



Campaign Management:
How many customers purchased more than $500 worth of alcohol in our Melbourne stores this year?

A wine export company…

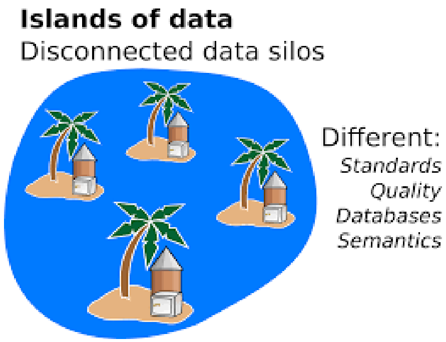
- We could write an SQL query:

```
Select      Name, Sum(Quantity), Sum(ItemCost) Sales
From        CustomerOrder a, OrderItem b, Wine c
Where       a.OrderCode = b.OrderCode
And         a.WineCode = c.WineCode
And         OrderDate = <today's date>
Group by    Name
```

- If executed at the end of day, it would return the value of all orders received for the day.
- If we were to execute the query everyday and append the results for each day in a table, we would be able to build up the historical information that we need.
- This is the beginning of a data warehouse for strategic information!

The Problem is...

- Too many databases
  - Everybody wanted one, or two, or more
  - Production, Marketing, Sales, Accounting ...
- Everybody got what database was best for them
  - IBM, Oracle, Access, Excel, file drawers, ...
- Data quality is poor
- Local optimisation → Global sub-optimisation
  - Organisation not able to capture benefits from its data and technology assets.
  - Islands of data, integrated data not available
- Data in wrong format for analysis



Operational vs Informational

OPERATIONAL DATABASES (OLTP)

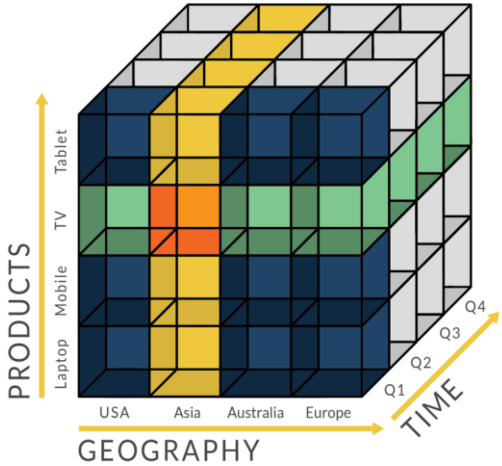
- Focus is on supporting day to day operations
  - Recording orders
  - Processing claims
  - Making shipments
  - Generating invoices
  - Receiving cash
  - Reserving airline seats

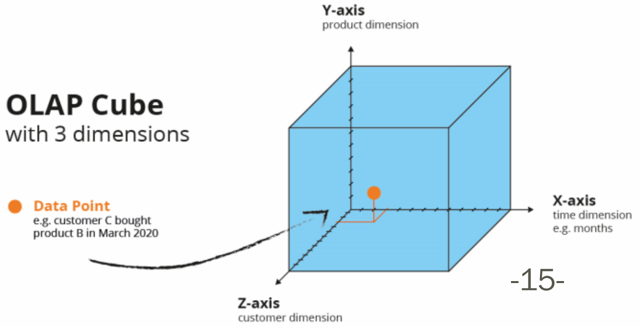
INFORMATION DATABASES (OLAP/BI)

- Have a different scope & different purpose
  - Show me the top products
  - Show me problem regions
  - Tell me why (drill down)
  - View other data (drill across)
  - Show the highest margins
  - Alert me if calls are high
- Focus is on getting information at a higher level

So we need a database that is…

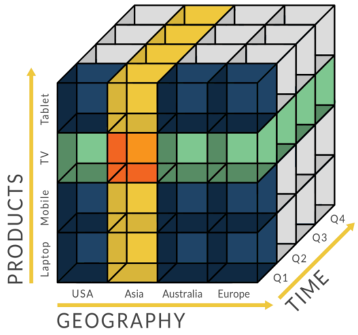
- Designed for analytic tasks
- Gets data from multiple locations
  - Internal / external
- Intuitive and easy to use
  - Allows direct access by users without IT support
- Conducive to long analysis sessions
- Read intensive
- Updated at known intervals and is stable
- Storing historical data also
- Able to allow users to run queries and get results on line
- Able to allow users to initiate reports



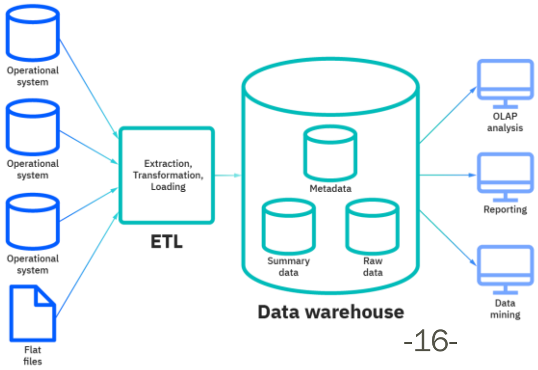


Our Solution

- The Data Warehouse!
- So, what's a data warehouse?
  - A single repository of organisational data
    - Current and historical
  - Integrates data from multiple sources
    - Internal and external
    - Extracts data from source systems, transforms, loads into the warehouse
    - "Single version of truth" – a holistic integrated view of organization data
  - Makes data available to managers/users
    - Without hindering day to day transactional work

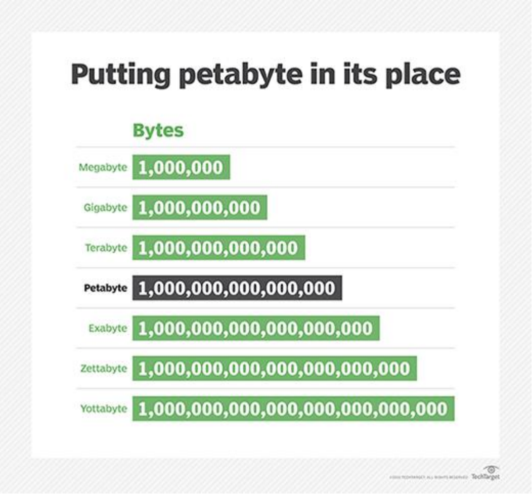


[Database VS Data Warehouse]



A Warehouse? An Analytical Database

- Data Warehouse
  - A single repository of organisational data
  - Integrates data from multiple sources
    - Extracts data from source systems, transforms, loads into the warehouse
    - “Single version of truth”
  - Makes data available to managers/users
  - Supports analysis and decision-making
- Involve a large data store (often several Terabytes, Petabytes, ...of data)



Comparison of databases: Decision Making

- **Transactional** databases (OLTP)
  - Used to answer operational questions
  - Primary data in operational databases
  - Large volumes of transactions with relatively small amounts of data per transaction
  - Some reporting requirements for operations

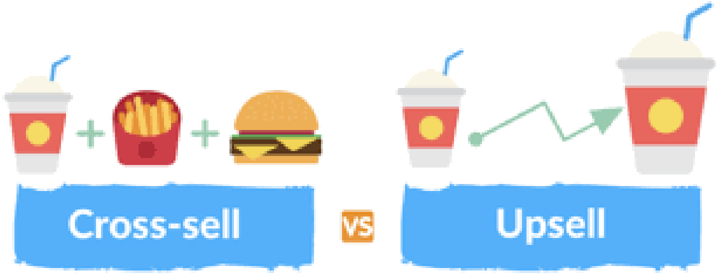
- **Analytical** databases (OLAP)
  - Used to answer strategic questions
  - Secondary data from operational databases
  - Substantial processing for transformations and integration
  - Large volumes of data for reporting

Why a Data Warehouse (DW)?

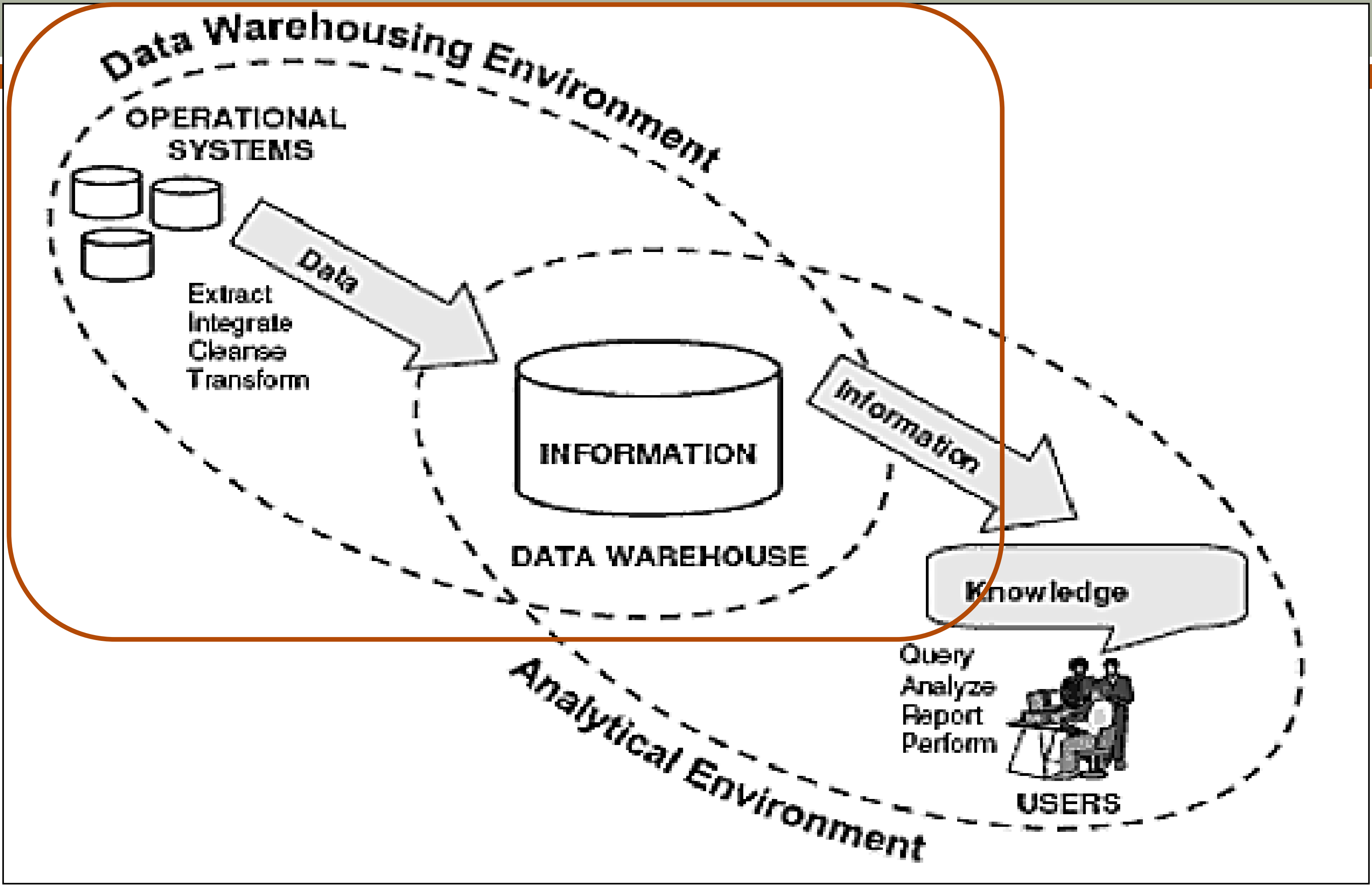
Modern Intelligent Enterprise > Relational DBMS

- Which items sell together? Which items to stock?
- Where and how to place the items? What discounts to offer?
- How best to target customers to increase sales at a branch?
- Which customers are most likely to respond to my next promotional campaign, and why?

[Why data warehouse? https://www.youtube.com/watch?v=KGHbY_Sales](https://www.youtube.com/watch?v=KGHbY_Sales)



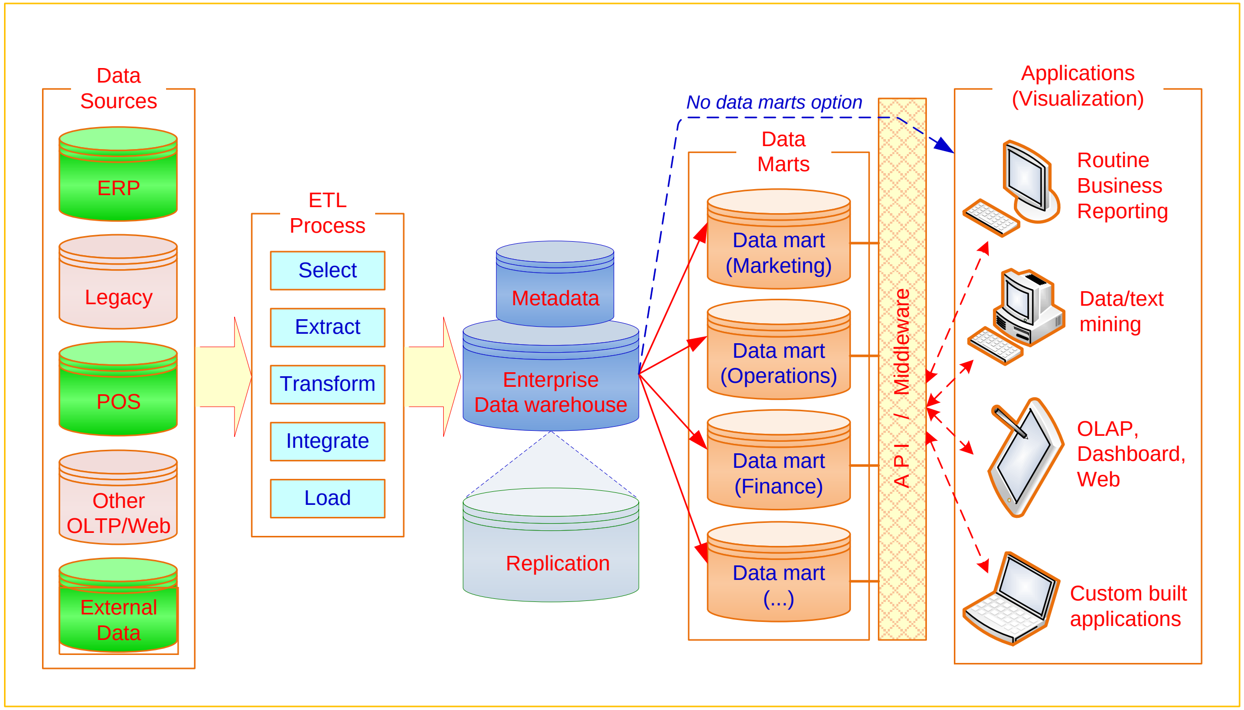
Data Warehousing and Analytical Environments



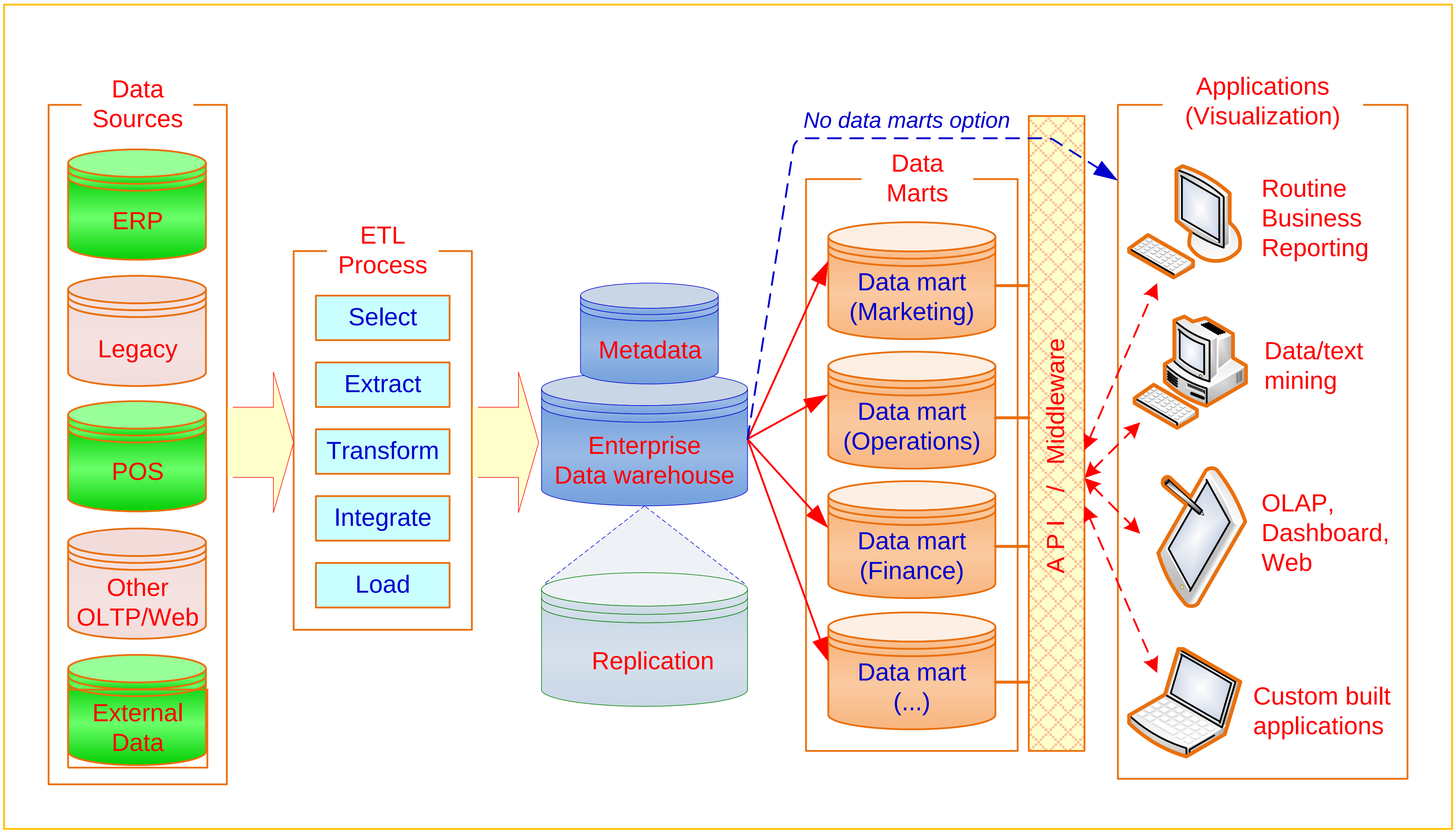
Database vs Data warehouse: https://www.youtube.com/watch?v=FxQG65OhXAQ

Basic Concept of Data Warehouse (DW)

1. A physical repository where relational data are <u>specially organised</u> to provide <u>enterprise-wide, cleansed data in a standardised format</u>
2. Data from multiple sources is extracted, transformed and loaded into the data warehouse. Then, analytics is performed using Online Analytical Processing (OLAP), which is based on the multidimensional data model.



A Generic Data Warehouse Framework



Definition of a Data Warehouse

Defined in many different ways:

- DW is a decision support database that is maintained *separately* from the organisation's operational database; support information processing by providing a solid platform of consolidated, historical data for analysis.

- "A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process." — William Bill Inmon

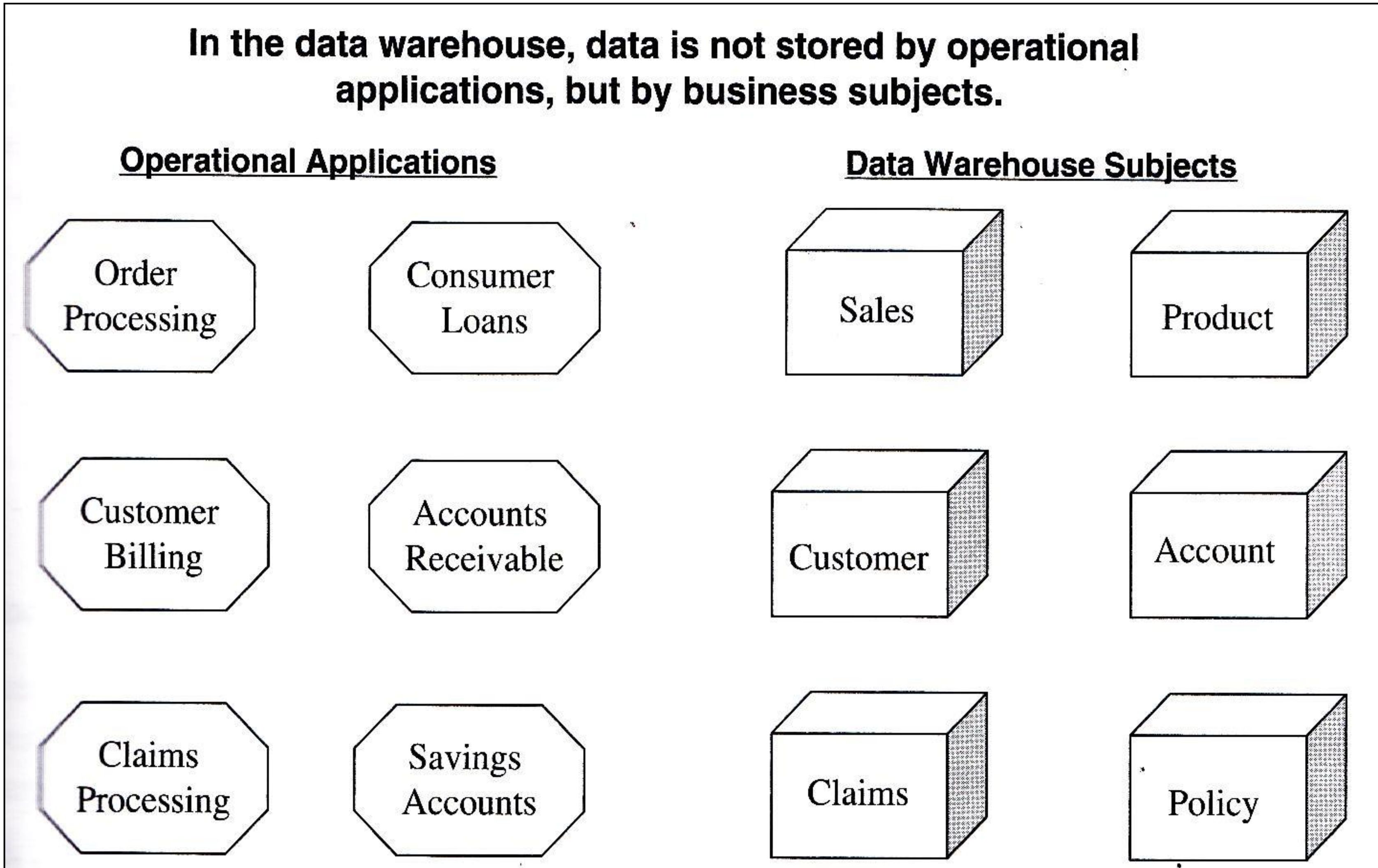
DW Definition

- Data Warehouse
  - A subject-oriented, integrated, time-variant, non-volatile collection of data used in support of management decision-making processes
    - **Subject-oriented:** e.g., customers, patients, students, products (see next slide)
    - **Integrated:** consistent naming conventions, formats, encoding structures; from multiple data sources
    - **Time-variant:** can study trends and changes
    - **Non-volatile/Non-updatable***:* read-only, periodically refreshed
- Data warehousing:
  - The **process** of constructing and using data warehouses

1. Subject-oriented Data

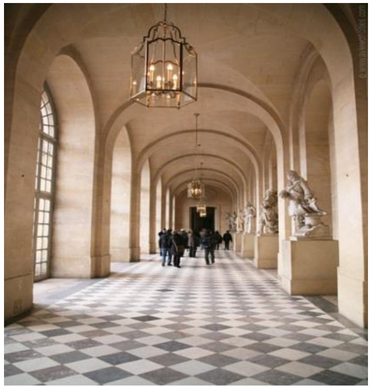
- Data warehouse organised around major <u>subjects</u> of the enterprise rather than the major application areas - ***<u>who</u>*** <u>**is involved,**</u> <u>not *what is done*</u>
  - who: customers, products, sales
  - what: customer invoicing, stock control, product sales
- This is reflected in the need to store decision-support data rather than application-oriented data
- E.g. Sales: "Who was our Top customer for Y item last year?"

1. Data Warehouse is Subject-Oriented



2. Integrated Data

- Corporate application-oriented data from different source systems is integrated
- This can include data that is inconsistent
  - For example, take-overs or mergers can result in different schemas for customers in different parts of the company
- The integrated data source must be made consistent to present a unified view of the data to the users
- Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources



Example of Data Integration

Cheque Account System

Jane Doe (name)

Female (gender)

Bounced cheque #145 on 1/5/95

Opened account 1994

← ***Operational data***

Savings Account System

Jane Doe

F (gender)

Opened account 1992

Investment Account System

Jane Doe

Owns 25 Shares Exxon

Opened account 1995

Customer

Jane Doe

Female

Bounced check #145

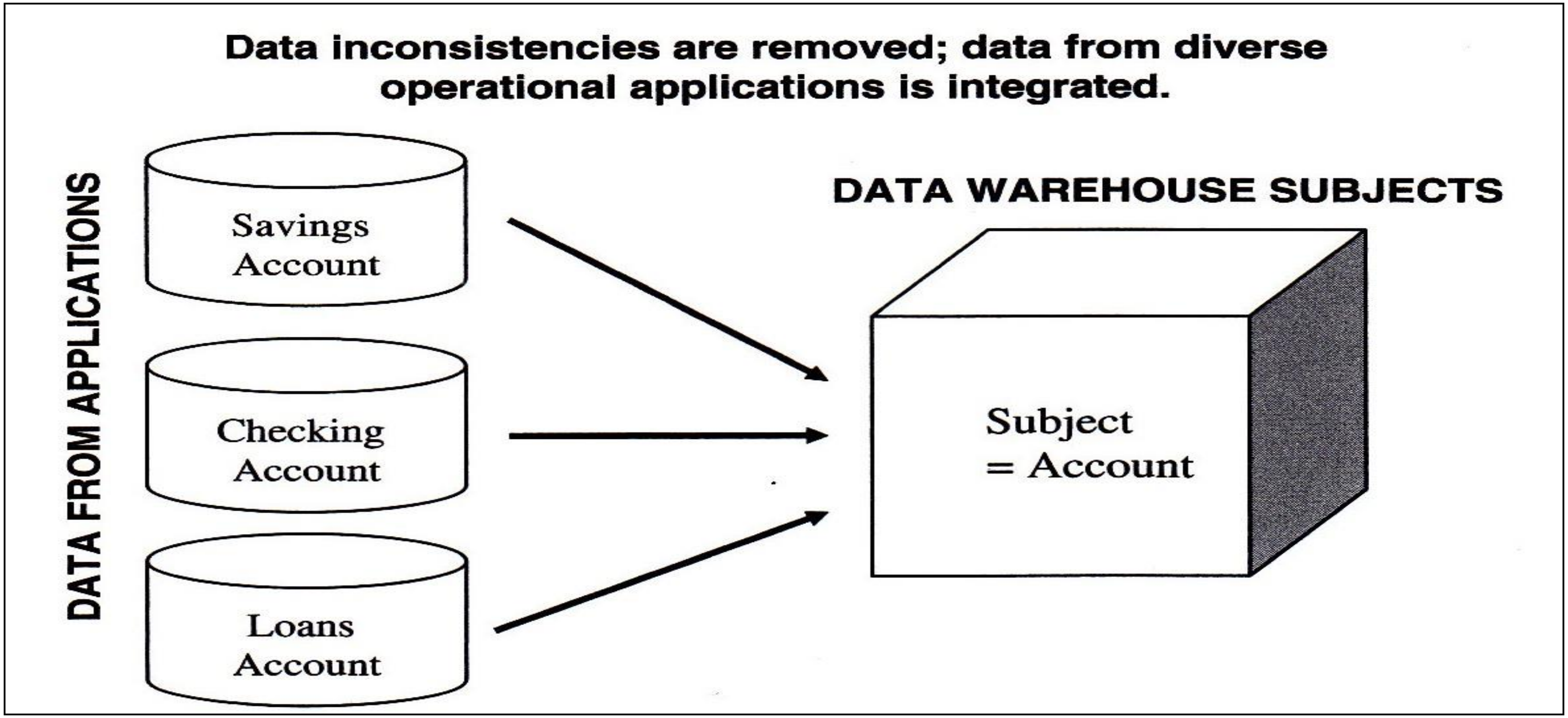
Married

Owns 25 Shares Exxon

Customer since 1992

↑ ***data warehouse***

Data Warehouse is Integrated



Some of the items that would need to standardise and make consistent: Naming Convention, Codes, Data Attributes, Measurements

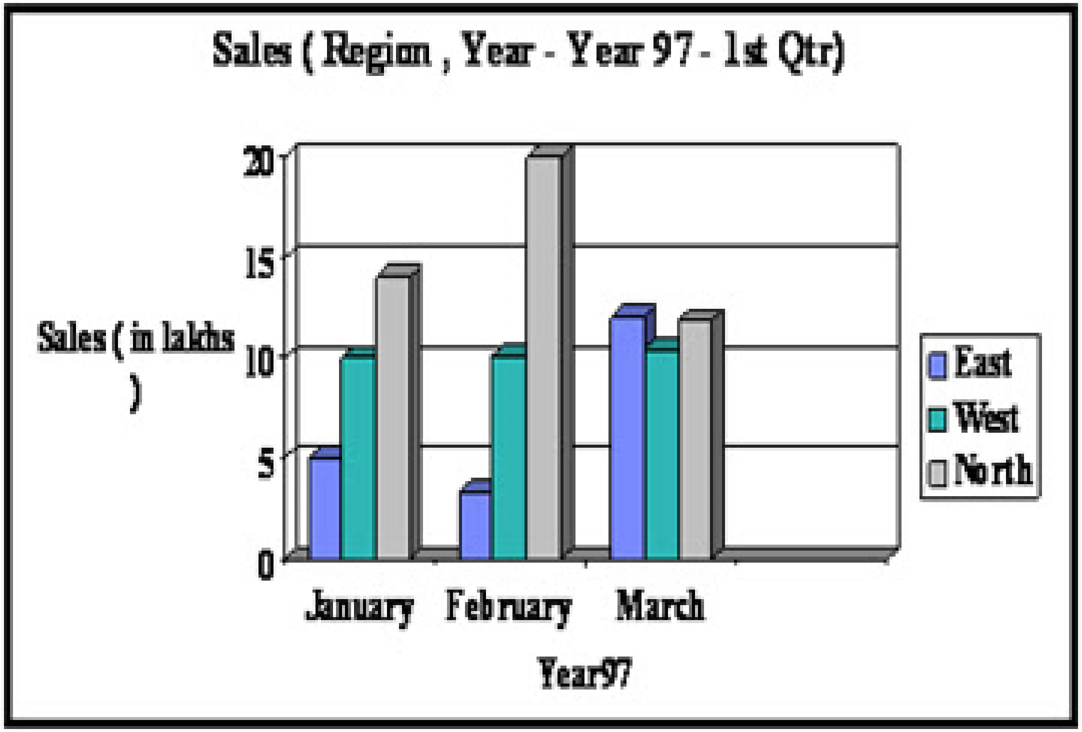
3. Time-Variant Data

- Focus on change over time
- Time-variance is shown in:
  - the data represents a series of snapshots
  - the extended time that the data is held,
  - the implicit or explicit association of time with all data,
  - Operational database: current value data
  - Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)

➔ **Focus on "Trends" / "change over time"**

Data Warehouse is Time-Variant

Current Data | **Historical Data**



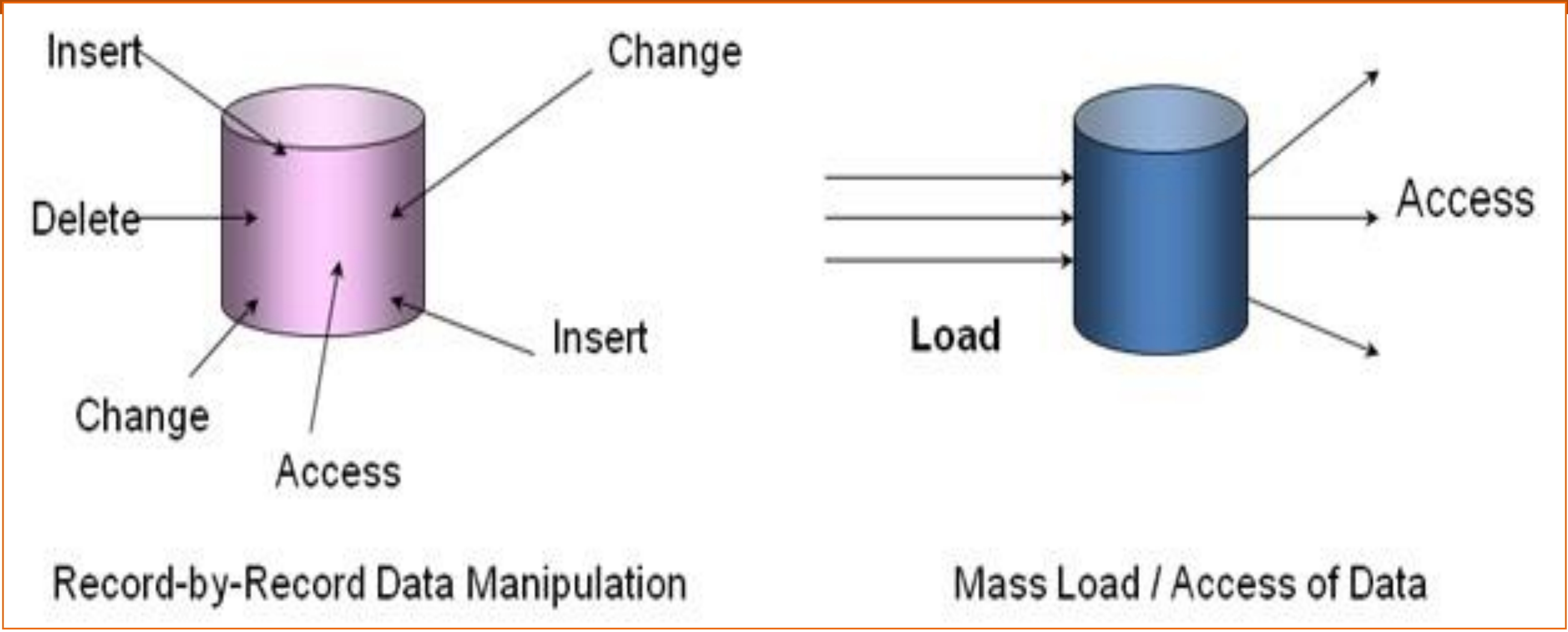
Transactional Storage

Data Warehouse

The time-variant nature of data in a data warehouse:

- Allows for analysis of the past
- Relates information to the present
- Enables forecast

4. Non-volatile/Non-updatable Data



- **Once loaded into the data warehouse, <u>the data is not updated</u>.**
- **Data acts as a <u>stable</u> resource for consistent reporting and comparative analysis.**
- **On the contrary, operational data is updated (inserted, deleted, modified).**

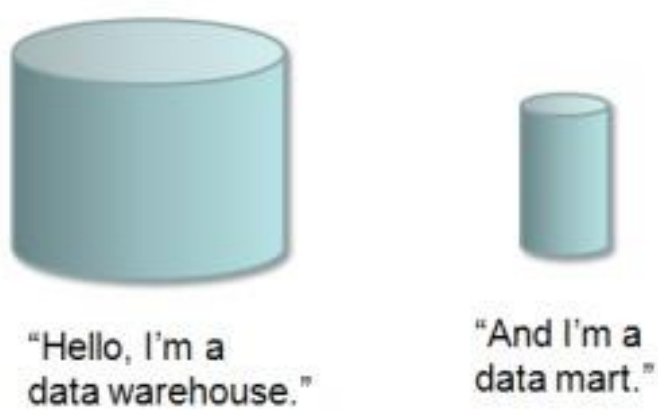
4. Non-volatile/Non-updatable Data

- Data in the warehouse is not updated in real-time but is refreshed from operational systems on a regular basis
  - Decision making depends on summaries of data rather than individual facts, so having totally up-to-date information is not so important
- New data is always added as a supplement to the warehouse, rather than a replacement
- Operational update of data does not occur in the data warehouse environment
- **data should not change; analyse what has occurred**

[Overview of DW: https://www.youtube.com/watch?v=zTs5zjSXnvs](https://www.youtube.com/watch?v=zTs5zjSXnvs)

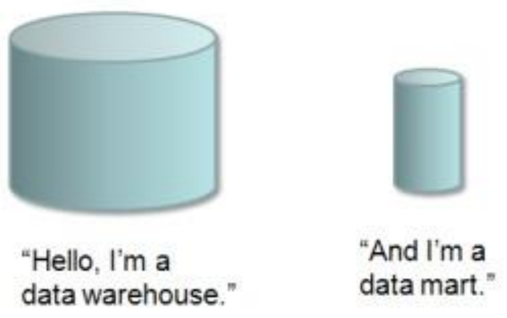
Data Mart

- Subsets of data warehouses that support the requirements of a particular department or business function
- Hold subset of the data, normally in summary form
- Can be stand-alone or linked to corporate data warehouse
- Data marts sometimes are popular because
  - Corporate-wide data warehouses can be difficult to build and used
  - More easily understood and navigated



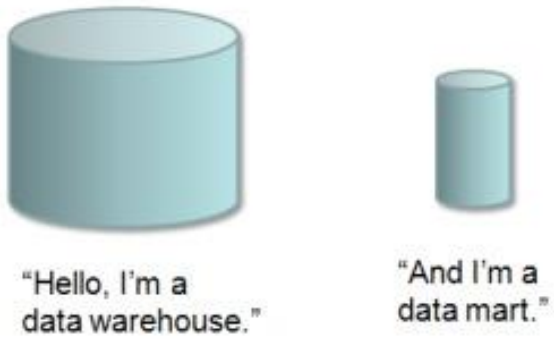
Reasons for Creating a Data Mart

1. To provide appropriately structured data as dictated by the requirements of the end-user access tools.
2. Building a data mart is **simpler** compared with establishing a corporate data warehouse.
3. The **cost** of implementing data marts is far less than that required to establish a data warehouse.
4. To provide data in a form that matches the <u>collective view</u> of a group of users
5. **Users** of a data mart are clearly defined and can be targeted for support **(Privacy)**



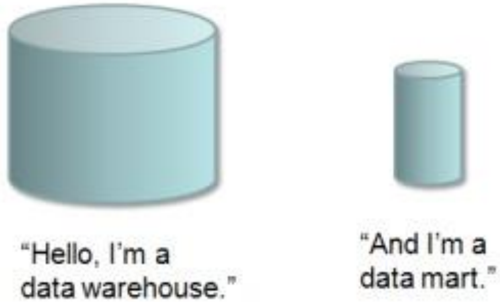
Data Warehouse Vs. Data Mart: Scope

- Data Warehouse
  - Application independent
  - Centralised, possibly enterprise-wide
  - Planned
- Data Mart
  - Specific DSS application
  - Decentralised by user area
  - Organic, possibly not planned



Data Warehouse Vs. Data Mart: Data, Subjects, and Sources

- Data Warehouse
  - Data is historical, detailed, and summarised
  - Data is lightly denormalized
  - Multiple subjects
  - Many internal and external sources
- Data Mart
  - Data has some history, is detailed and summarised
  - Data is highly denormalized
  - One central subject or concern to users
  - Few internal and external sources

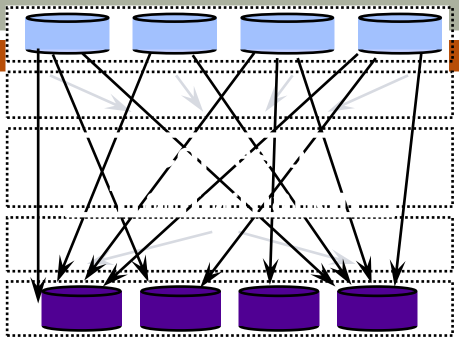


Comparing Enterprise-level DW (EDW) and Data Mart

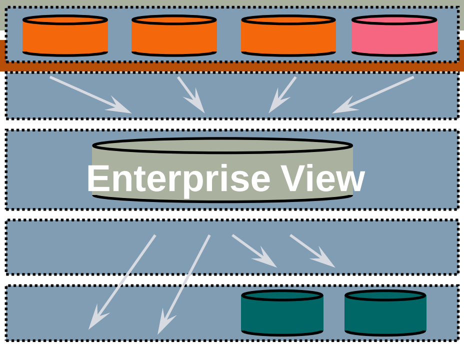
TABLE 3.3 Contrasts between the DM and EDW Development Approaches

Effort	DM Approach	EDW Approach
Scope	One subject area	Several subject areas
Development time	Months	Years
Development cost	\$10,000 to \$100,000+	\$1,000,000+
Development difficulty	Low to medium	High
Data prerequisite for sharing	Common (within business area)	Common (across enterprise)
Sources	Only some operational and external systems	Many operational and external systems
Size	Megabytes to several gigabytes	Gigabytes to petabytes
Time horizon	Near-current and historical data	Historical data
Data transformations	Low to medium	High
Update frequency	Hourly, daily, weekly	Weekly, monthly
Technology		
Hardware	Workstations and departmental servers	Enterprise servers and mainframe computers
Operating system	Windows and Linux	Unix, Z/OS, OS/390
Databases	Workgroup or standard database servers	Enterprise database servers
Usage		
Number of simultaneous users	10s	100s to 1,000s
User types	Business area analysts and managers	Enterprise analysts and senior executives
Business spotlight	Optimizing activities within the business area	Cross-functional optimization and decision making

Type of Data Mart



Independent Data Marts



Dependent Data Marts

Advantages
- Fast implementation
- Quick ROI
- Departmental control
- Not reliant on IT for data

Disadvantages
- Multiple data models / data duplication
- No consistent corporate data model
- Multiple interfaces to manage/maintain
- No single version of the truth

Advantages
- Single Version of the Truth
- Clean / Scrubbed / Quality data
- Consistent data model
- Robust data transformation
- Flexible data mart options

Disadvantages
- Requires a data warehouse
- Must align with corporate strategy

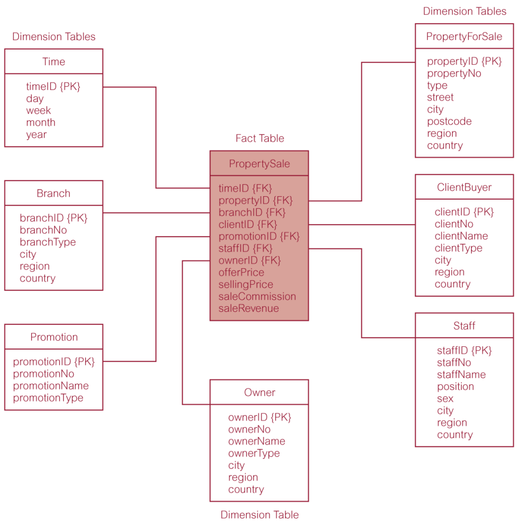
Dimensional Model

❑ A dimensional model is a data modelling technique used in data warehousing that organises data into easily understandable structures.

❑ It is designed to facilitate fast and efficient data retrieval and analysis, particularly for reporting and business intelligence purposes.

❑ In a dimensional model, data is organised around ‘facts’ (i.e., numerical measures) and ‘dimensions’ (i.e., descriptive characteristics).

❑ **Facts** represent the measures of interest, such as sales revenue or units sold, while **dimensions** represent the various categories or attributes of the data, such as product, location, time, and customer.



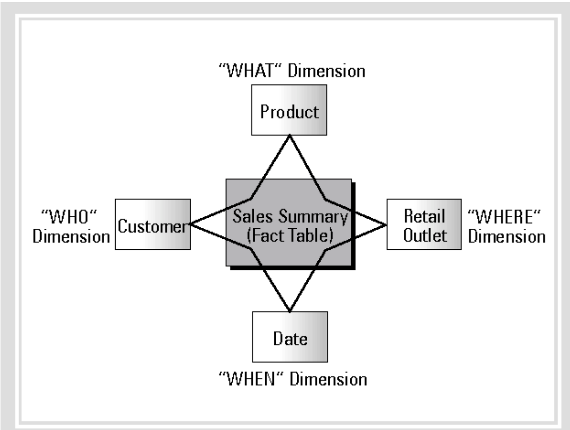


Figure 1. *Star Schema Example*

Dimensional model: Schema types

- The dimensional model is often represented visually as a star schema or snowflake schema, with the fact table at the centre and the dimension tables branching out from it.
- This structure allows for easy navigation and aggregation of data, as well as efficient querying and reporting.
- Dimensional modelling is widely used in data warehousing and business intelligence applications, as it provides a flexible and efficient way to organise and analyse large volumes of data.
- Most common types of schema are:
  - Star
  - Snowflake
  - Starflake

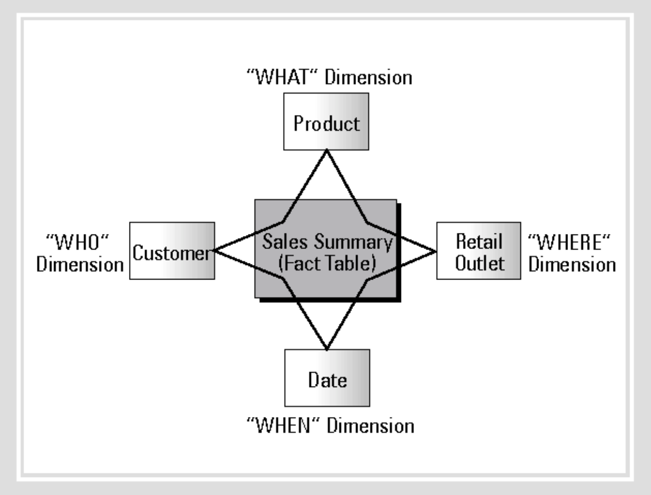
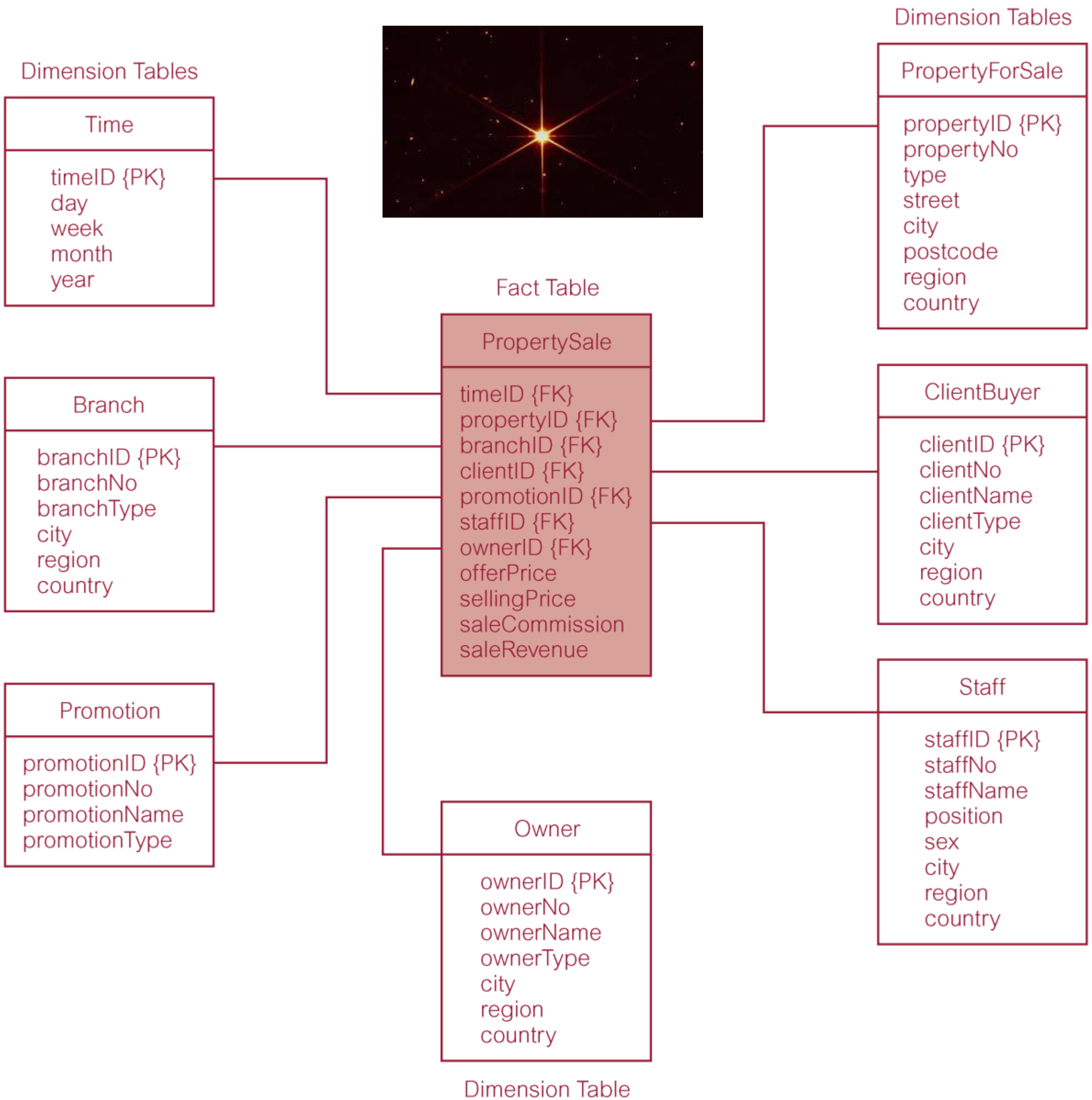
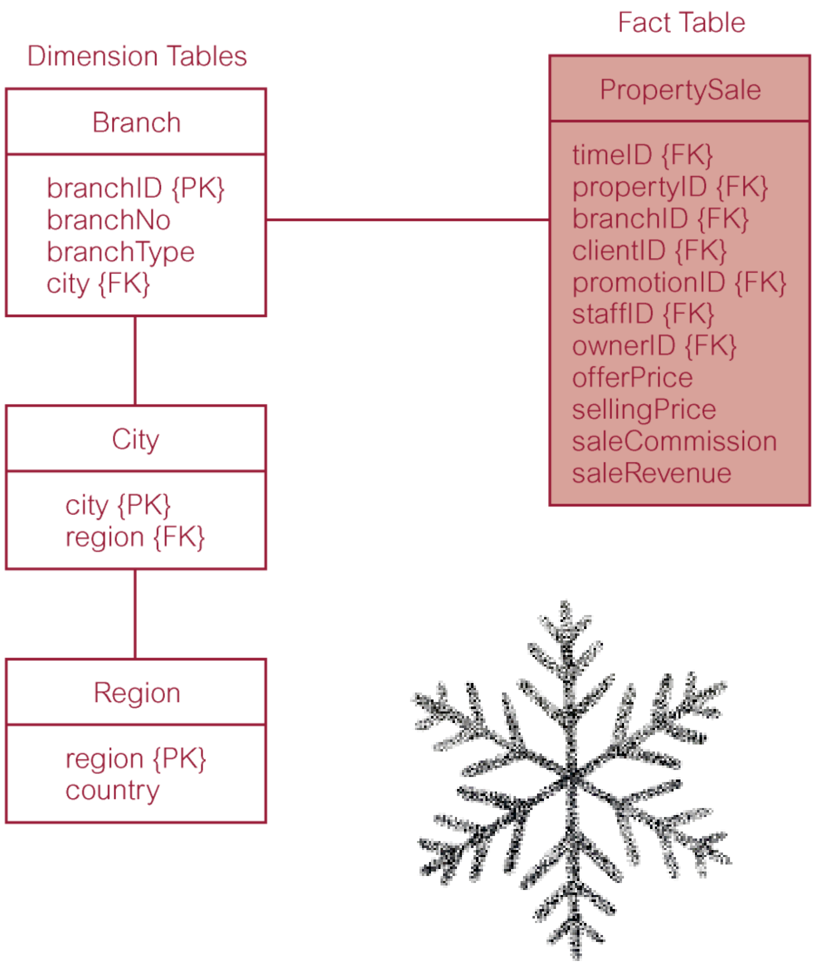


Figure 1. *Star Schema Example*

Dimensional Model using Star schema: property sales



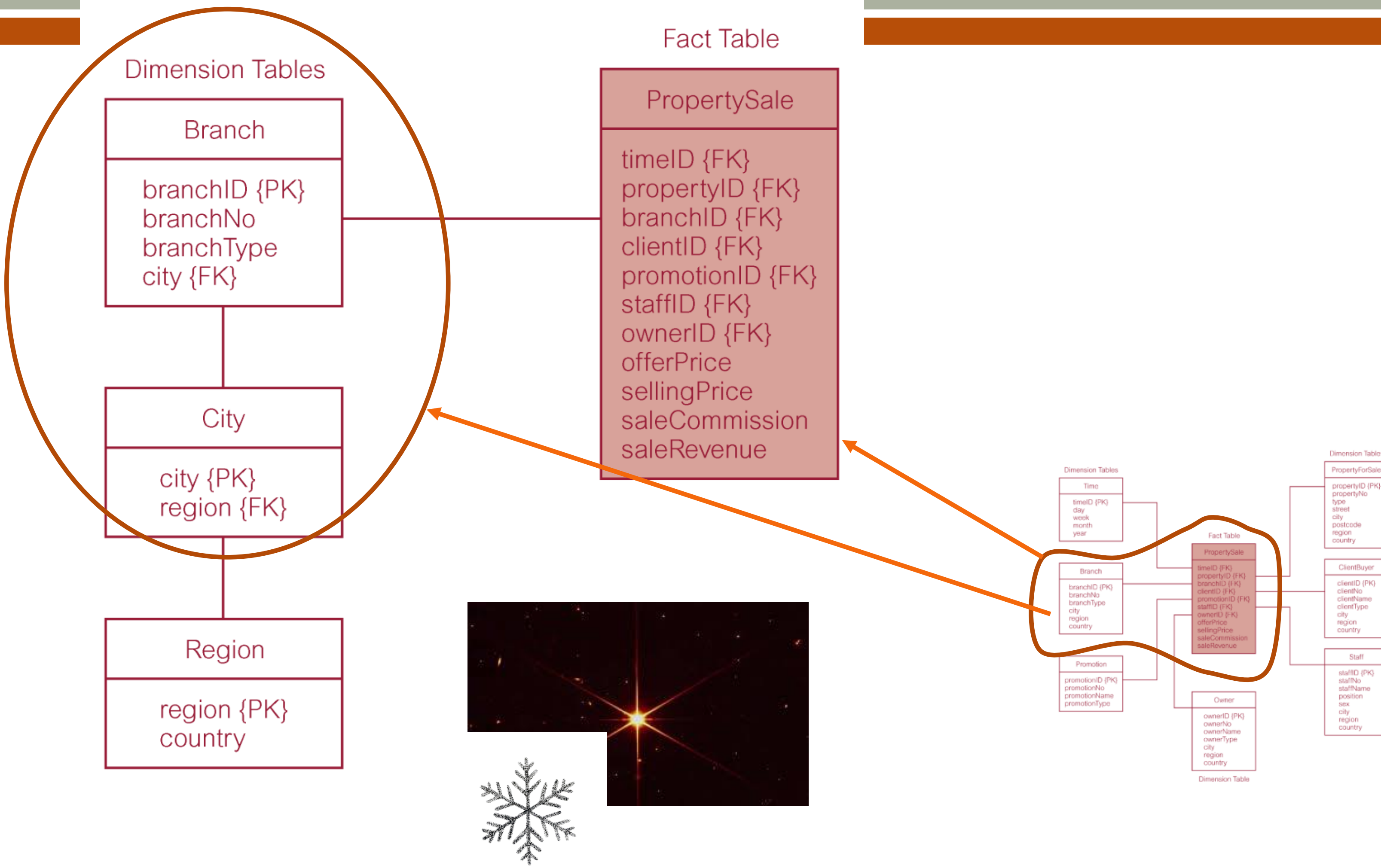
Dimensional Model using snowflake schema: property sales



Representation of Data in DW

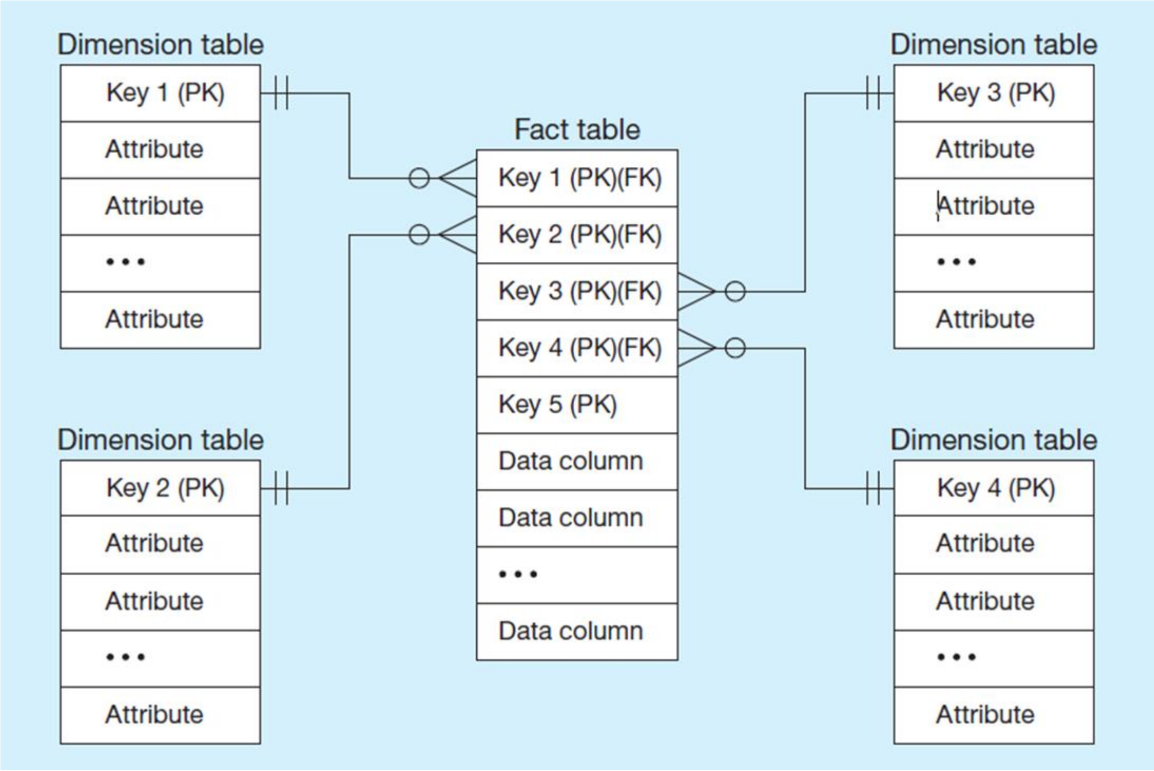
- Star schema (‘dimensional model’)
  - The most commonly used and the simplest style of dimensional modeling
  - Contain a fact table surrounded by and connected to several dimension tables
- Snowflakes schema
  - An ***extension*** of star schema where the diagram resembles a snowflake in shape

Dimensional Model using Starflake schema: property sales



Components of a Star Schema

- **Fact** tables contain factual or quantitative data
- Each dimension table has a primary key that appears as a foreign key in the fact table, whose primary key is a concatenation of all of the foreign keys.
- Dimension tables contain descriptions about the subjects of the business



Step-by-step guide: Convert relational tables to a dimensional model

1. Step 1: Identify the Business Process. This will help you determine which tables to use as the fact table and dimension tables.
2. Step 2: Identify the Fact Table. Identify the fact table by looking for the table that contains the measurable data for the business process. This might be a table that contains transactional data, such as sales, orders, or shipments.
3. Step 3: Identify the Dimensions. Identify the dimensions by looking for the related tables that provide descriptive data about the fact table data. These might include tables that contain information about products, customers, locations, or time.
4. Step 4: Create the Fact Table. Create a new fact table based on the identified fact table in the relational model. The fact table should contain the same primary key as the original fact table, as well as any foreign keys needed to link to the dimensions.
5. Step 5: Create the Dimension Tables. Create dimension tables for each of the identified dimensions. Each dimension table should contain the primary key of the dimension, as well as any descriptive attributes associated with the dimension.
6. Step 6: Link the Dimension Tables to the Fact Table. Add foreign key columns to the fact table to link it to the dimension tables. Each foreign key column should correspond to a primary key in a dimension table.
7. Step 7: Add Hierarchies to the Dimension Tables. Add hierarchical structures to the dimension tables, if applicable. For example, a time dimension table might have a hierarchy that includes year, quarter, month, and day.

Video: Facts and Dimensions

An Example: Retail Trading

- A large grocery store with approx. 500 stores
- Each store has approx. 60,000 products on shelves
- Need to maximise profit and keep shelves stocked
- Important decisions concern pricing and promotion
- Promotion types are:
  – Temporary price reductions
  – Newspaper advertisements
  – Shelf and end-aisle displays
  – Coupons

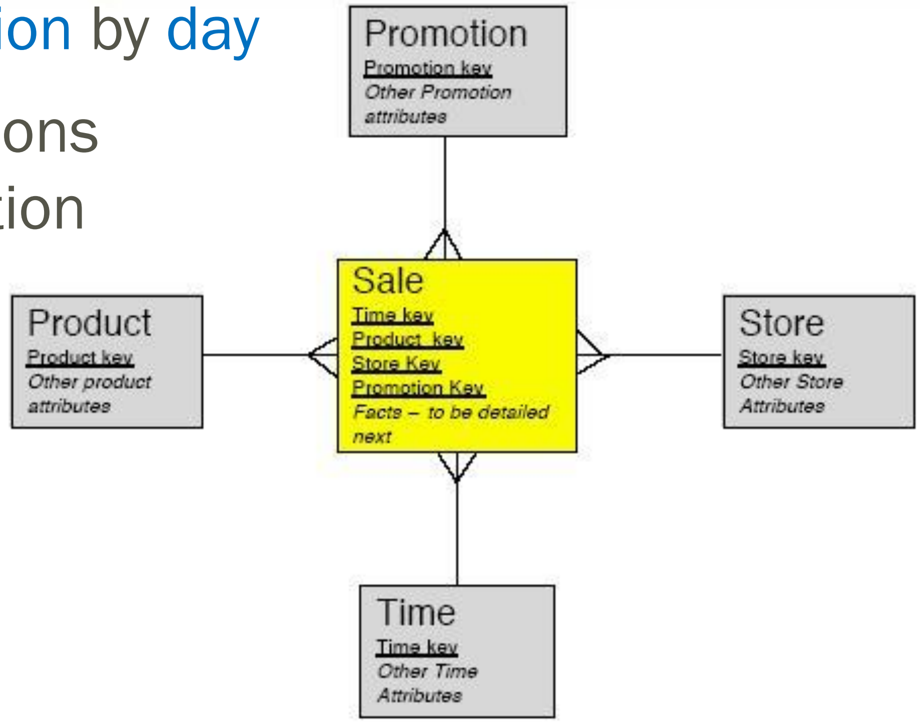
Steps in the dimensional model design process: Retail Trading

<u>Key Steps:</u>

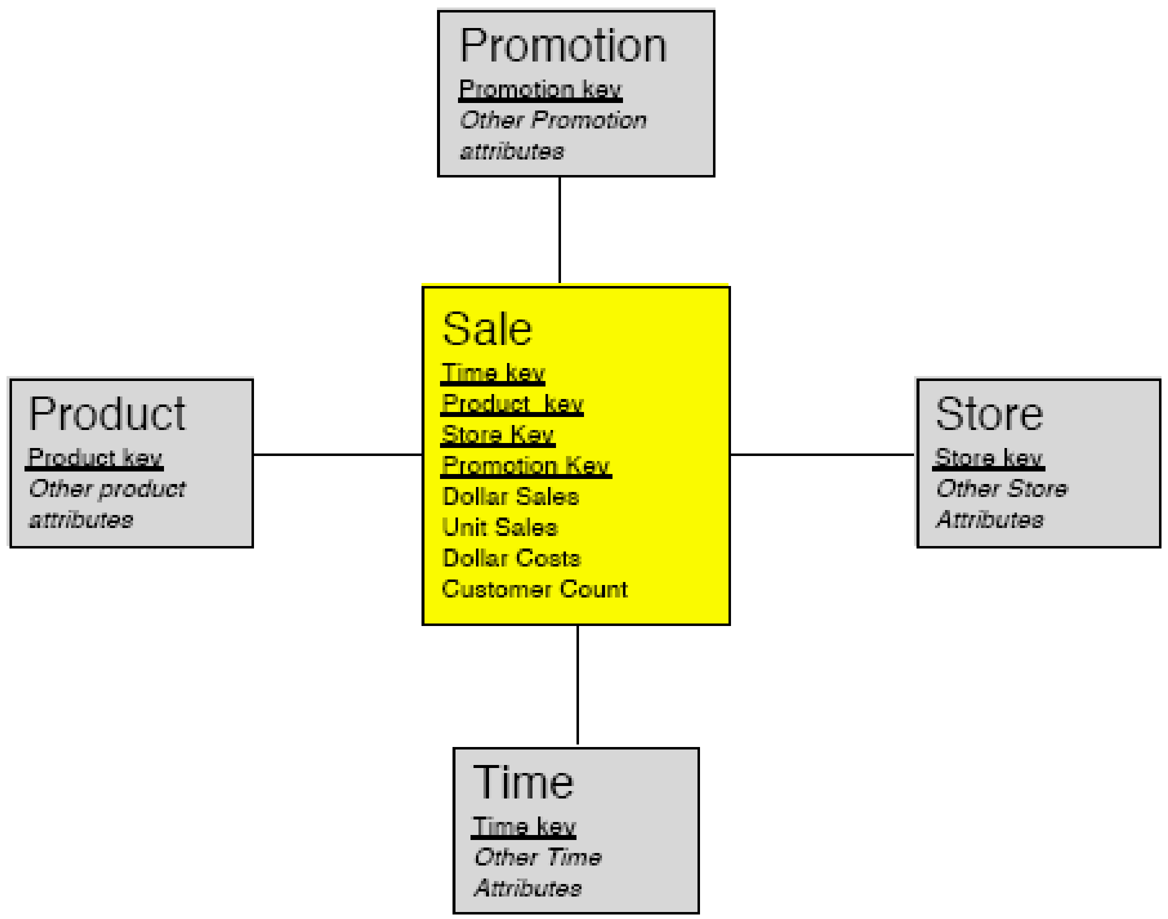
1. Choose a Business Process
   - Daily Item Movement

2. Choose & Complete the grain of the fact table
   - Product SKU by store by promotion by day

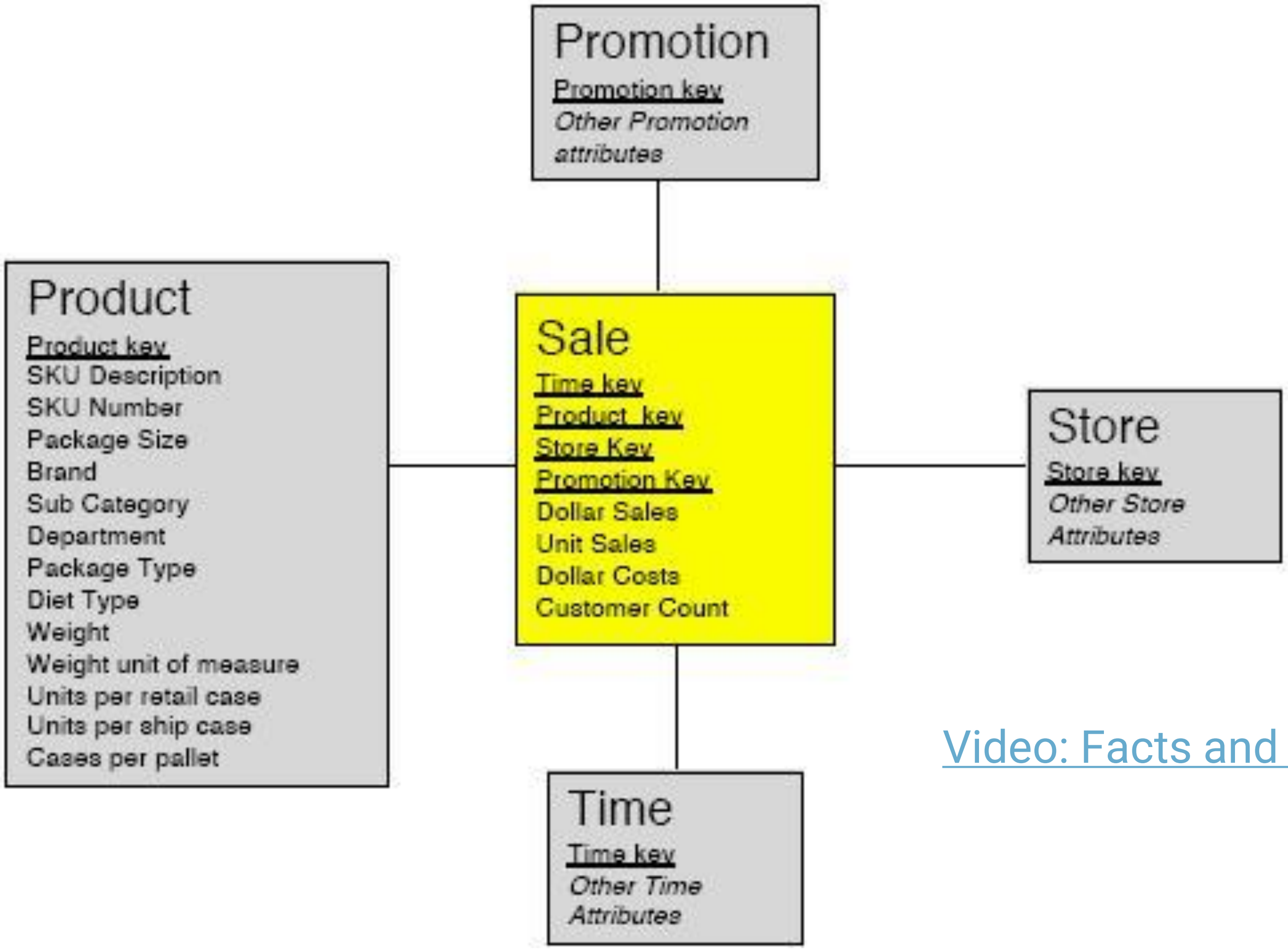
3. Choose & Complete the Dimensions
   - Time, product, store and promotion



Step 2: Complete the measured facts

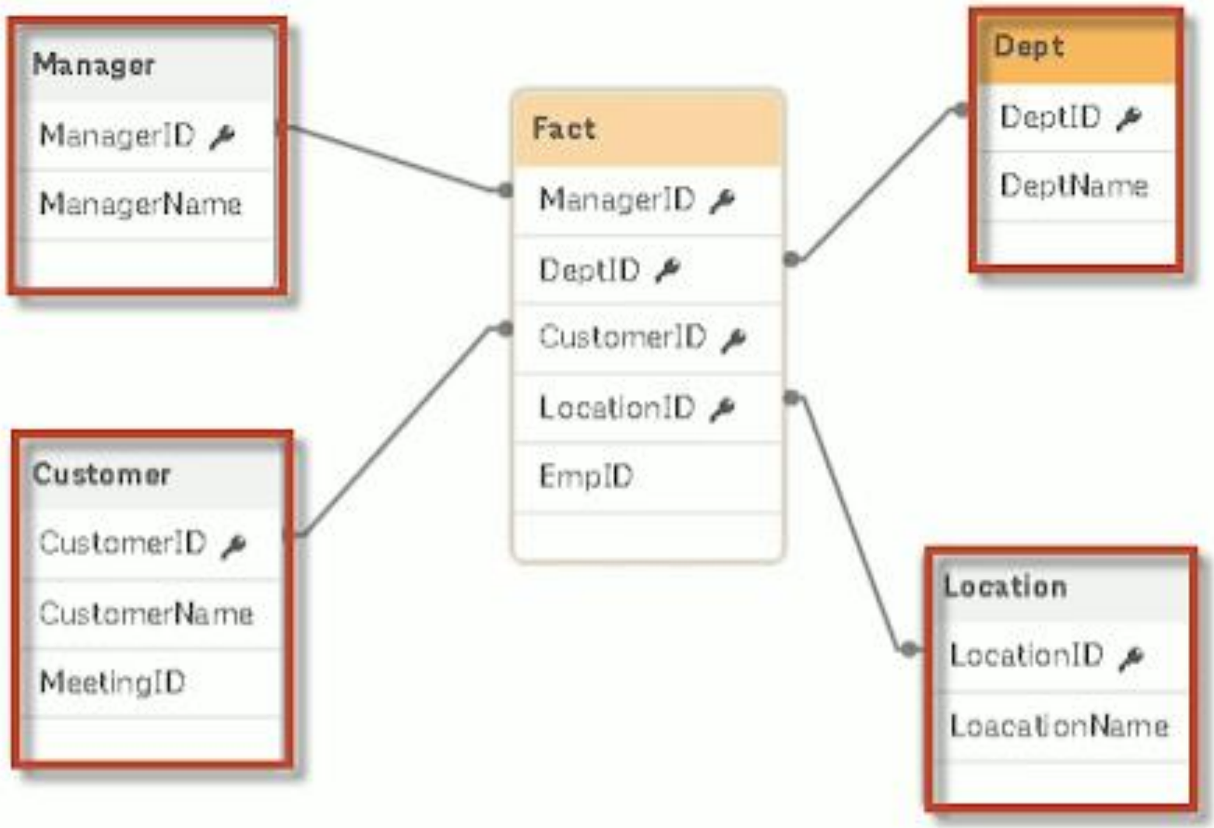


Step 3: Complete the dimension tables

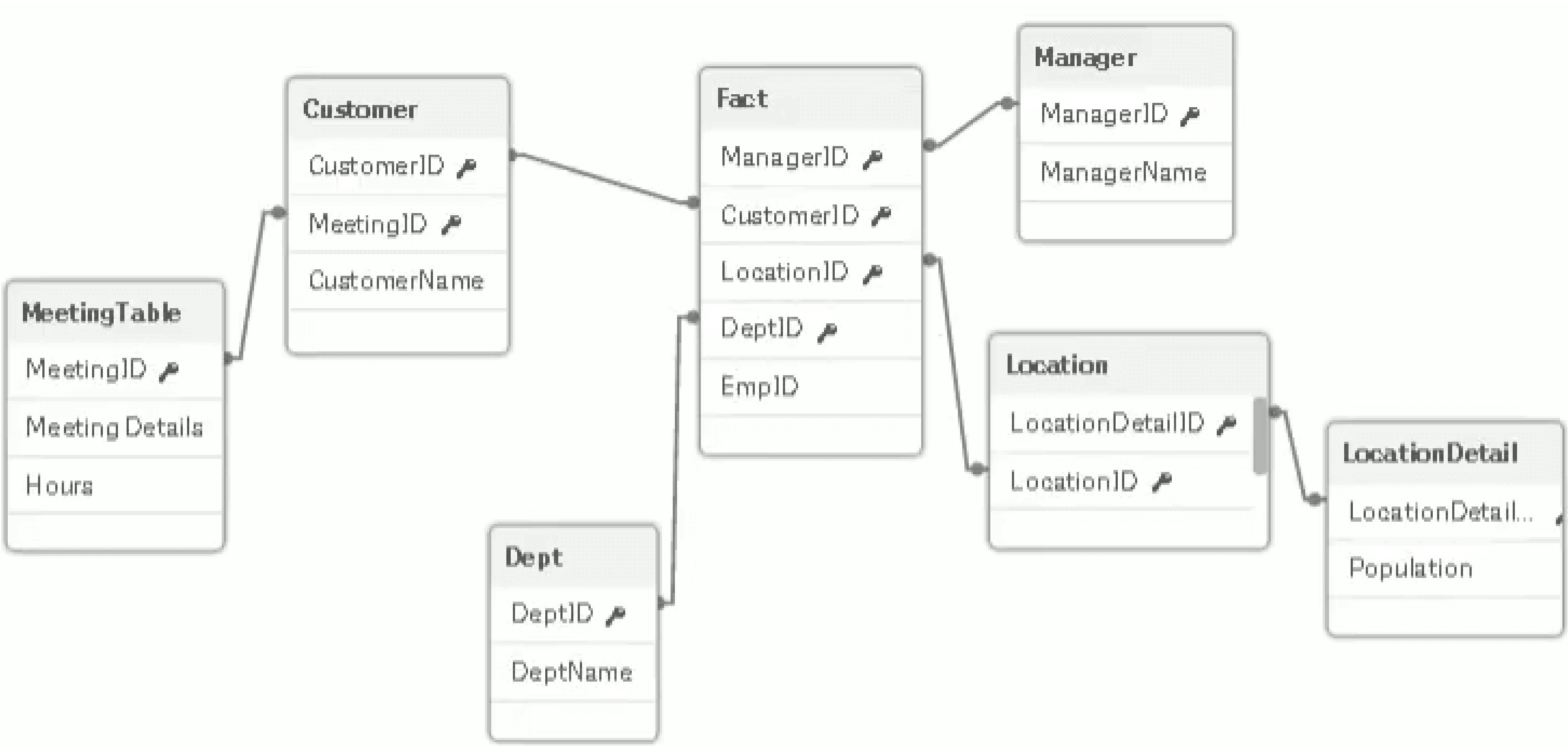


Video: Facts and Dimensions

Exercise: Design a snowflake schema based on this star schema

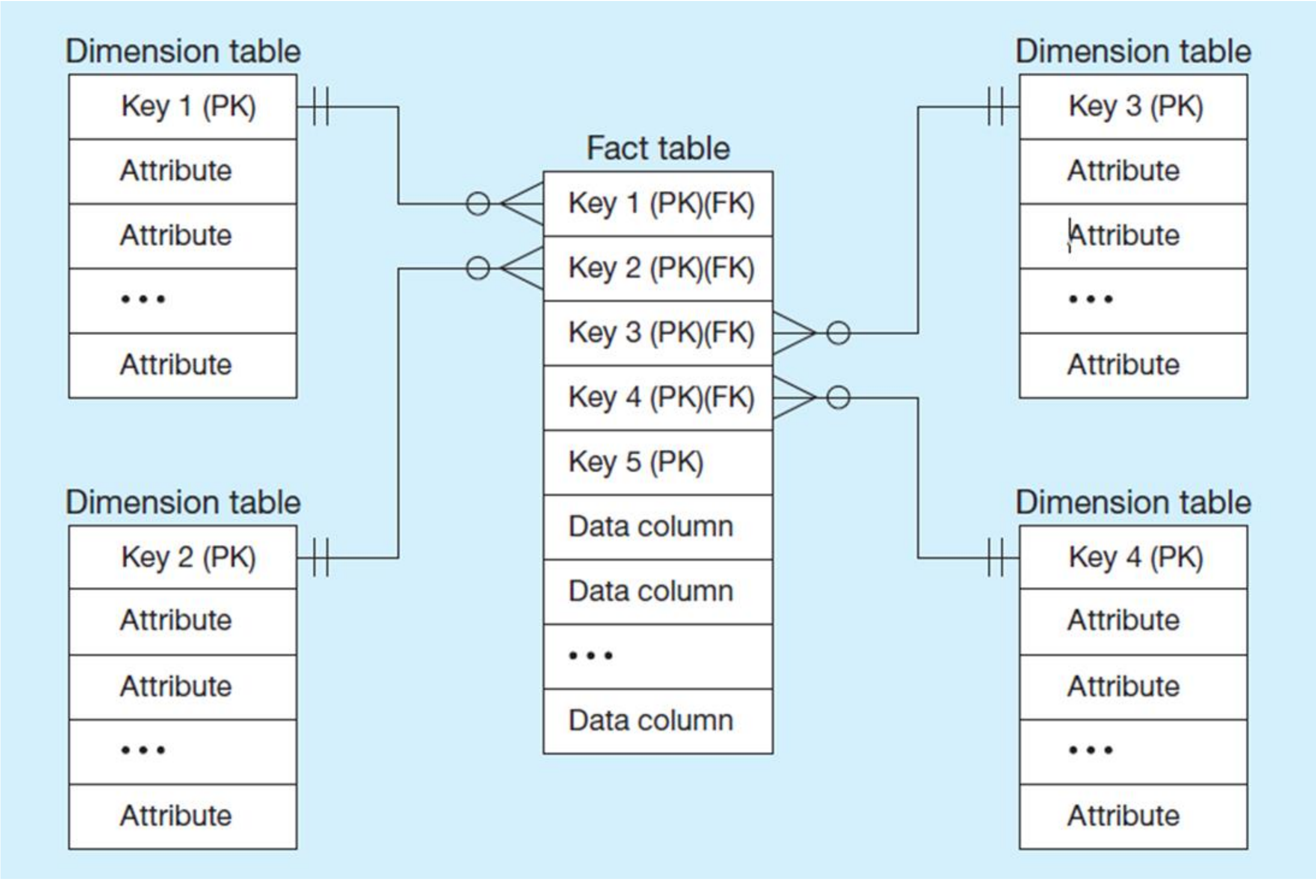


Exercise: Design a snowflake schema based on this star schema



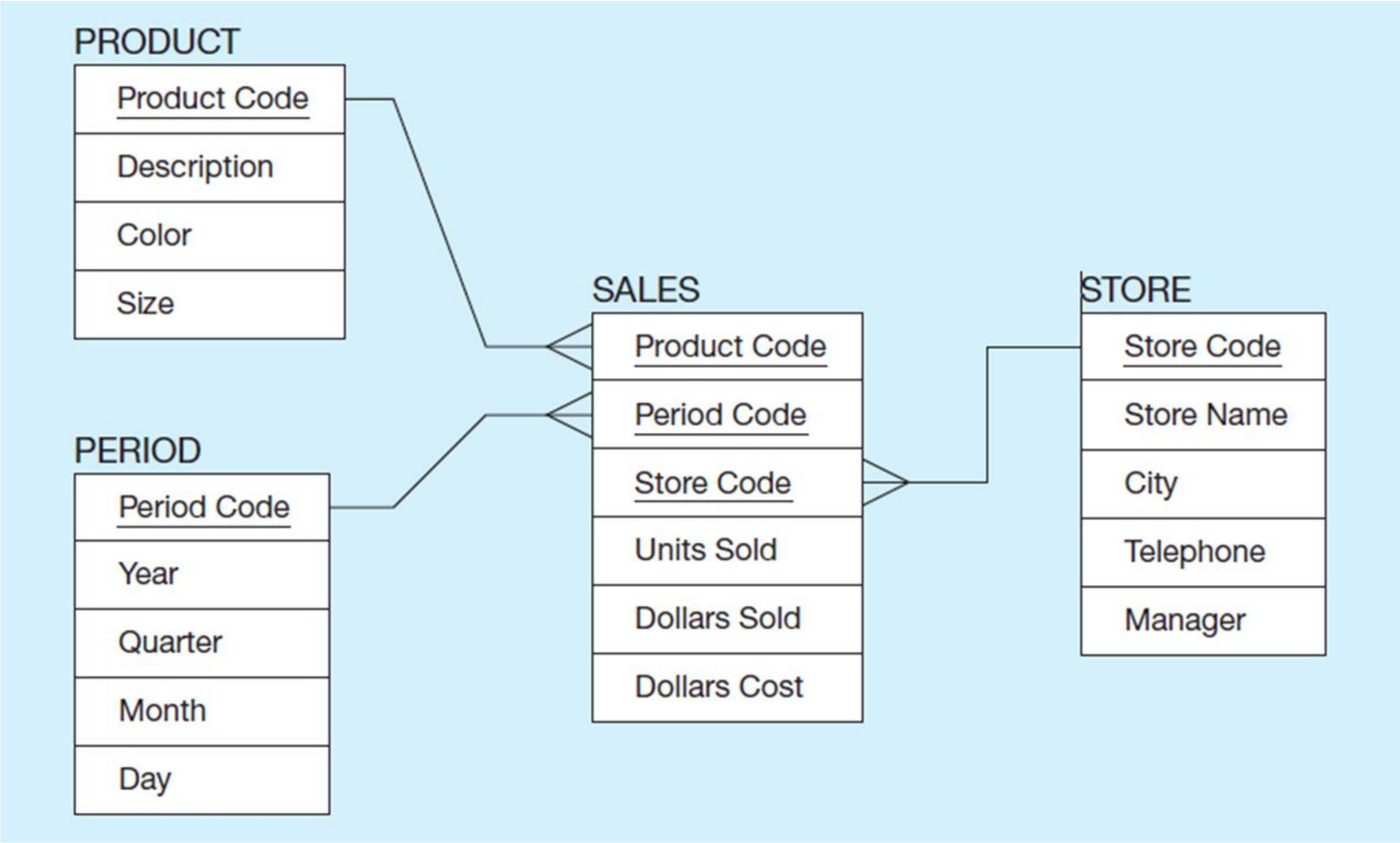
Components of a Star Schema

- Fact tables contain factual or quantitative data
- Dimension tables contain descriptions about the subjects of the business
- Excellent for ad-hoc queries, but bad for online transaction processing



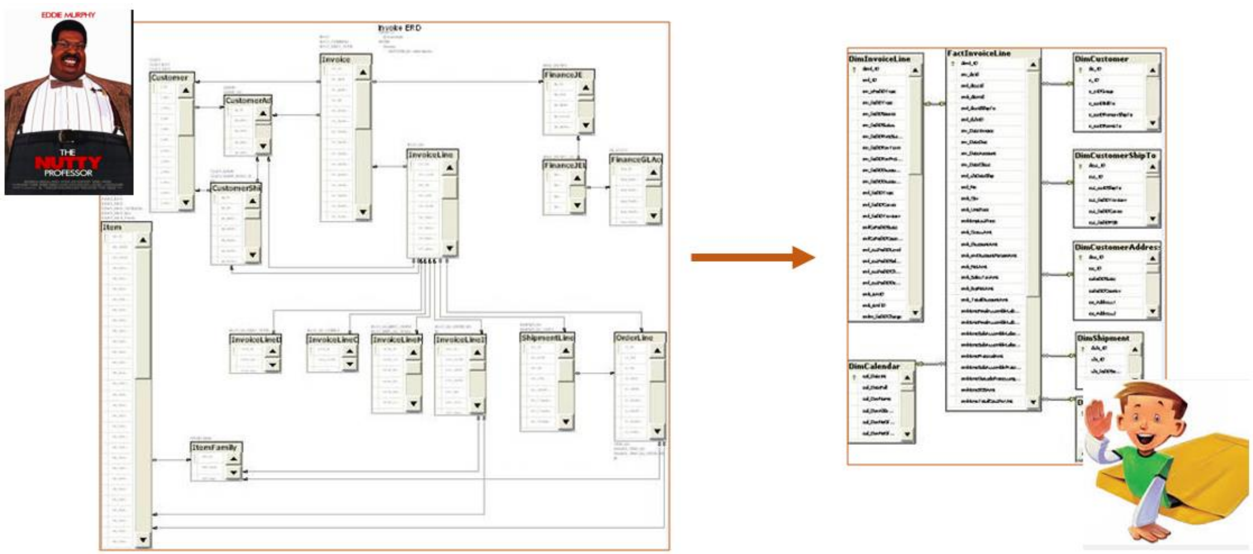
Star Schema Example

- Fact table provides statistics for sales broken down by product, period, and store dimensions



Dimension Tables: Denormalisation

- Dimension tables in star schemas are **denormalised** resulting in:
  - Fewer tables
  - Simpler for users to navigate
  - Reduced number of complex multi-join tables.



16 3NF tables ....became ....8 2NF tables

Source: https://www.delorabradish.com/modeling-for-bi/denormalizing-snowflake-3nf-dimension-tables

Surrogate Keys

- Dimension table keys should be **surrogate** (non-intelligent and non-business related), because:
  - Business keys may change over time
  - Helps keep track of non-key attribute values for a given production key
  - Surrogate keys are simpler and shorter
  - Surrogate keys can be same length and format for all keys

https://www.zoossa.com.au/orphaned-tree-kangaroo-saved-in-world-first/

Analysis of Data in DW: OLAP

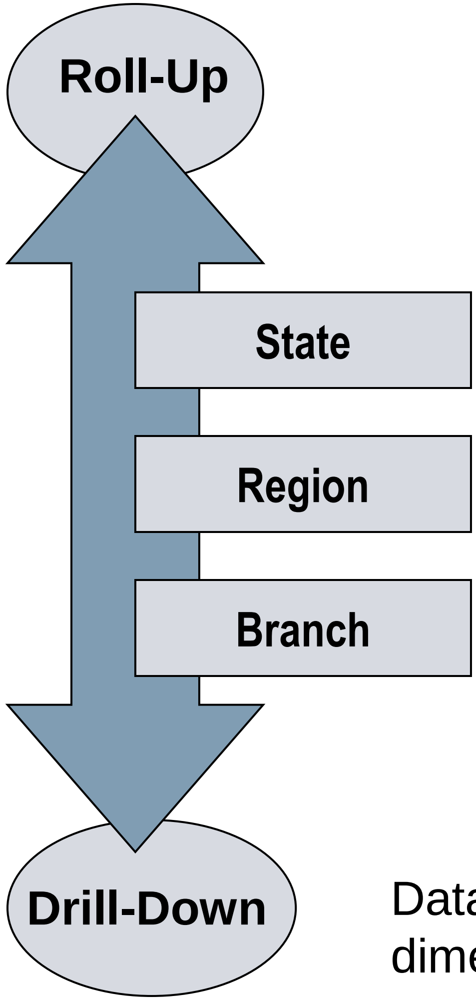
- OLTP vs. OLAP…
- OLTP (Online Transaction Processing)
  - Capturing and storing data from ERP, CRM, POS, …
  - The main focus is on efficiency of routine tasks

- **OLAP (Online Analytical Processing)**
  - Converting data into information for decision support
  - Data cubes, drill-down / roll-up, slice & dice (see next slide)
  - Requesting ad hoc reports
  - Conducting statistical and other analyses
  - Developing multimedia-based applications

OLAP (Online Analytical Processing)

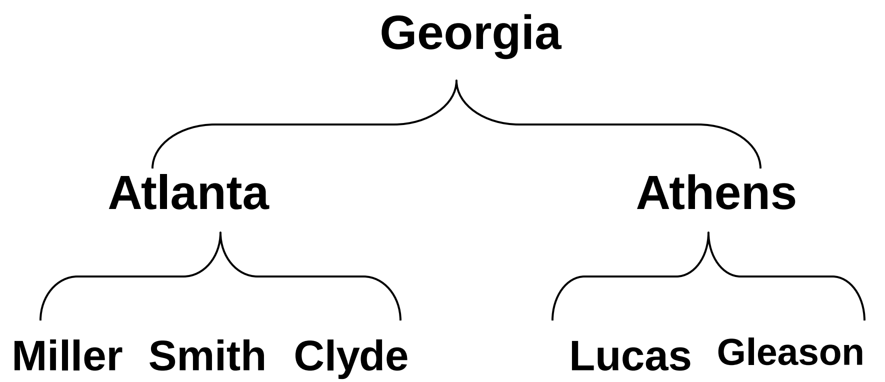
- Multi-dimensional OLAP supports analytical operations
  - Consolidation/Roll-up
    - aggregation of data such as simple “roll-up”
    - computing all of the data relationships for one or more dimensions
    - e.g. branch offices can be rolled-up to cities
  - Drill-down
    - reverse of consolidation
    - involves displaying data at a more detailed level
    - navigating among levels of data ranging from the most summarized (up) to the most detailed (down)
  - Slicing and dicing
    - refers to the ability to look at the data from different viewpoints.
    - Slice - a subset of a multidimensional array.
    - Dice - a slice on more than two dimensions

OLAP Features: Drill-Down and Roll-Up



Productivity by Manager

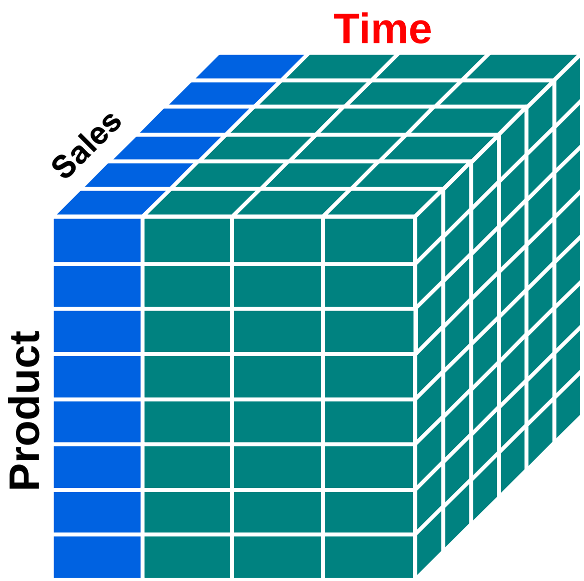
- three level hierarchy -



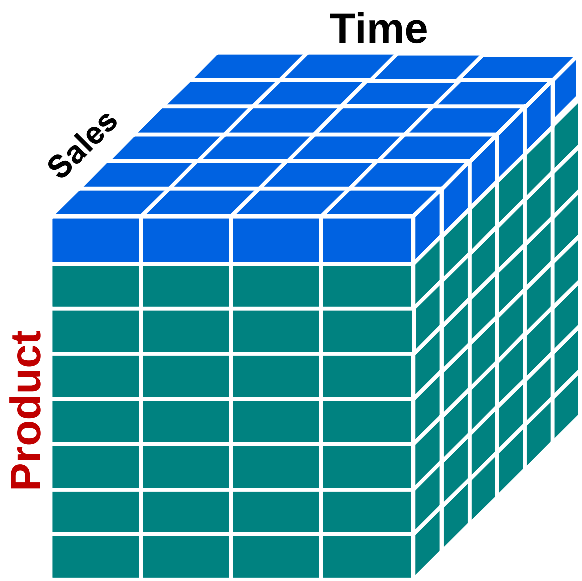
Data can be disaggregated and aggregated along a dimension according to their natural hierarchy

OLAP Features: Slice

"Slice" to view different perspectives

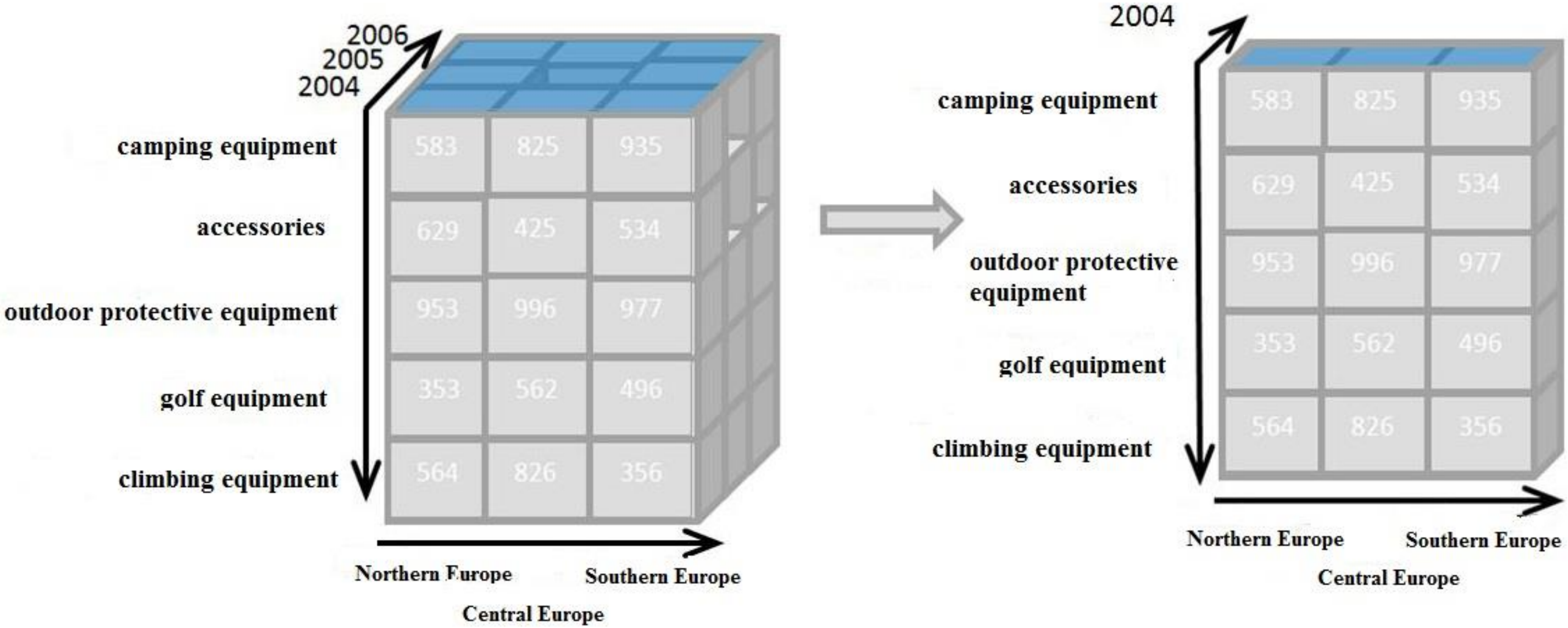


by Time (January)
by all Products
by all Sales



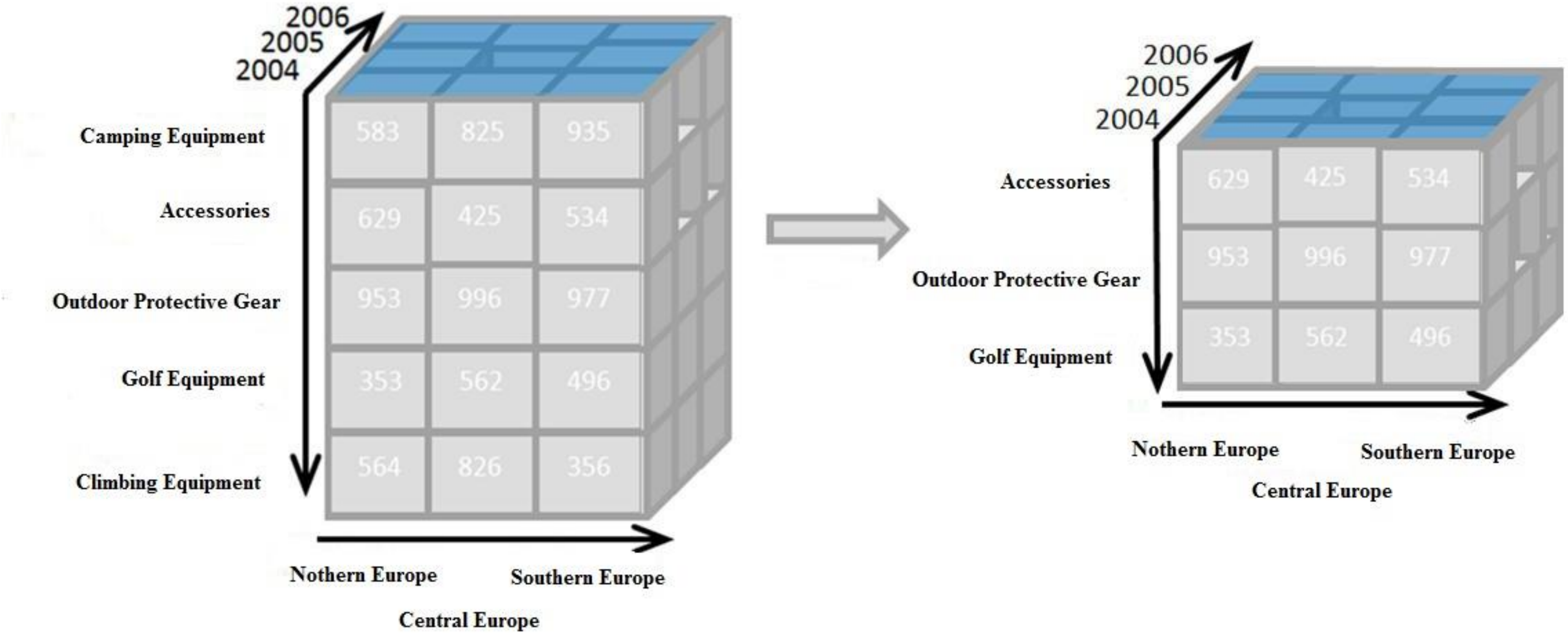
by Product X
by all time
by all Sales

Slice (subset)



Source: https://pediaa.com/what-is-the-difference-between-slice-and-dice-in-data-warehouse/

Dice (subcube)



Source: https://pediaa.com/what-is-the-difference-between-slice-and-dice-in-data-warehouse/

Slice vs Dice

SLICE IN DATA WAREHOUSE

VERSUS

DICE IN DATA WAREHOUSE

SLICE IN DATA WAREHOUSE	DICE IN DATA WAREHOUSE
Act of picking a rectangular subset of a cube by choosing a single value for one of its dimensions, creating a new cube with fewer dimensions	Act of producing a subcube by allowing the analyst to pick specific values of multiple dimensions
Used to select one particular dimension from a given cube and to provide a new subcube	Used to select two or more dimensions from a given cube and to provide a new subcube

Summary of OLAP: https://www.youtube.com/watch?v=2ryG3Jy6eIY

Representing Multi-Dimensional Data

- Example of two-dimensional query (3 fields)
  - “What is the total revenue generated by property sales in each city, in each quarter of 2014?”

- Choice of representation is based on types of queries end-user may ask
- Compare representation - **three-field relational table** versus **two-dimensional matrix**

3-Field Table Versus 2-Dimensional Matrix

City	Time	Total Revenue
Glasgow	Q1	29726
Glasgow	Q2	30443
Glasgow	Q3	30582
Glasgow	Q4	31390
London	Q1	43555
London	Q2	48244
London	Q3	56222
London	Q4	45632
Aberdeen	Q1	53210
Aberdeen	Q2	34567
Aberdeen	Q3	45677
Aberdeen	Q4	50056
....	....	....
....	....	....

City

Time

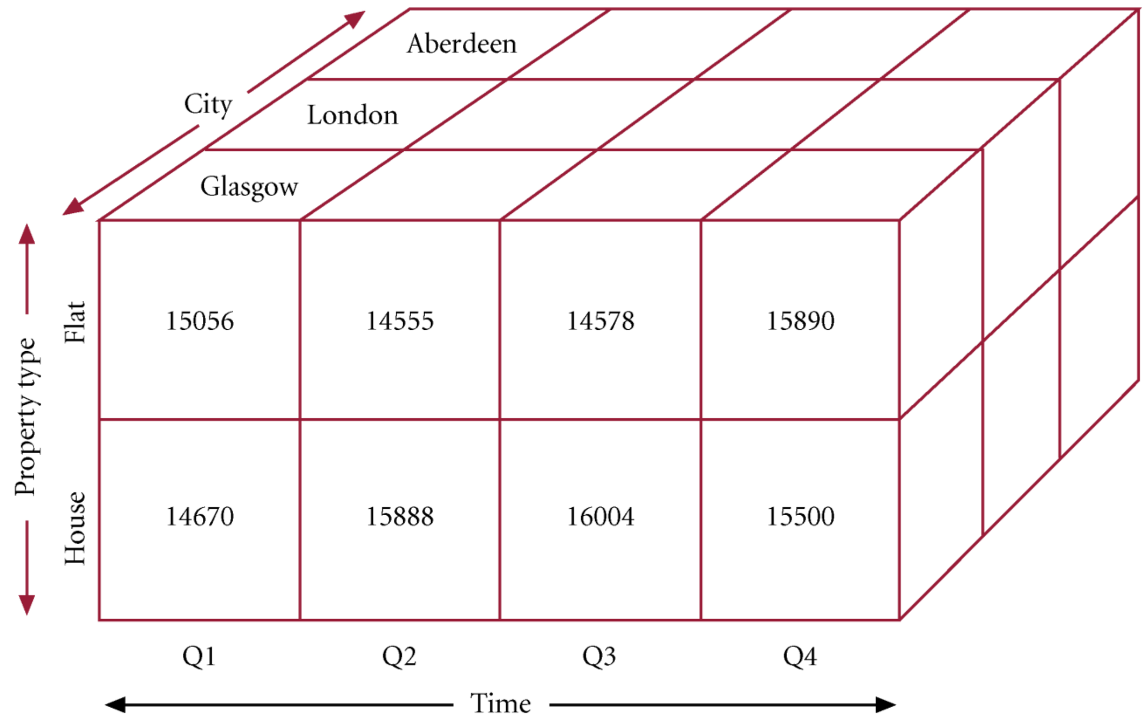
City / Quarter	Glasgow	London	Aberdeen	....
Q1	29726	43555	53210	....
Q2	30443	48244	34567	....
Q3	30582	56222	45677	....
Q4	31390	45632	50056	....

Representing Multi-Dimensional Data

- Example of three-dimensional query
  - ‘What is the total revenue generated by property sales for each type of property (Flat or House) in each city, in each quarter of 2014?”
- Compare representation - **four-field relational table** versus **three-dimensional cube**

4-Field Table Versus 3-Dimensional Cube

Property Type	City	Time	Total Revenue
Flat	Glasgow	Q1	15056
House	Glasgow	Q1	14670
Flat	Glasgow	Q2	14555
House	Glasgow	Q2	15888
Flat	Glasgow	Q3	14578
House	Glasgow	Q3	16004
Flat	Glasgow	Q4	15890
House	Glasgow	Q4	15500
Flat	London	Q1	19678
House	London	Q1	23877
Flat	London	Q2	19567
House	London	Q2	28677
....	....	....	....
....	....	....	....



Representation of Multi-dimensional Data

- Dimensional hierarchy and pre-aggregation can significantly reduce size of the cube and the need to calculate values ‘on-the-fly’. Thus significantly speeding up execution of multi-dimensional queries.

- Majority of multi-dimensional queries use **summarised**, high-level data. Solution is to pre-aggregate (consolidate) all logical subtotals and totals along all dimensions.

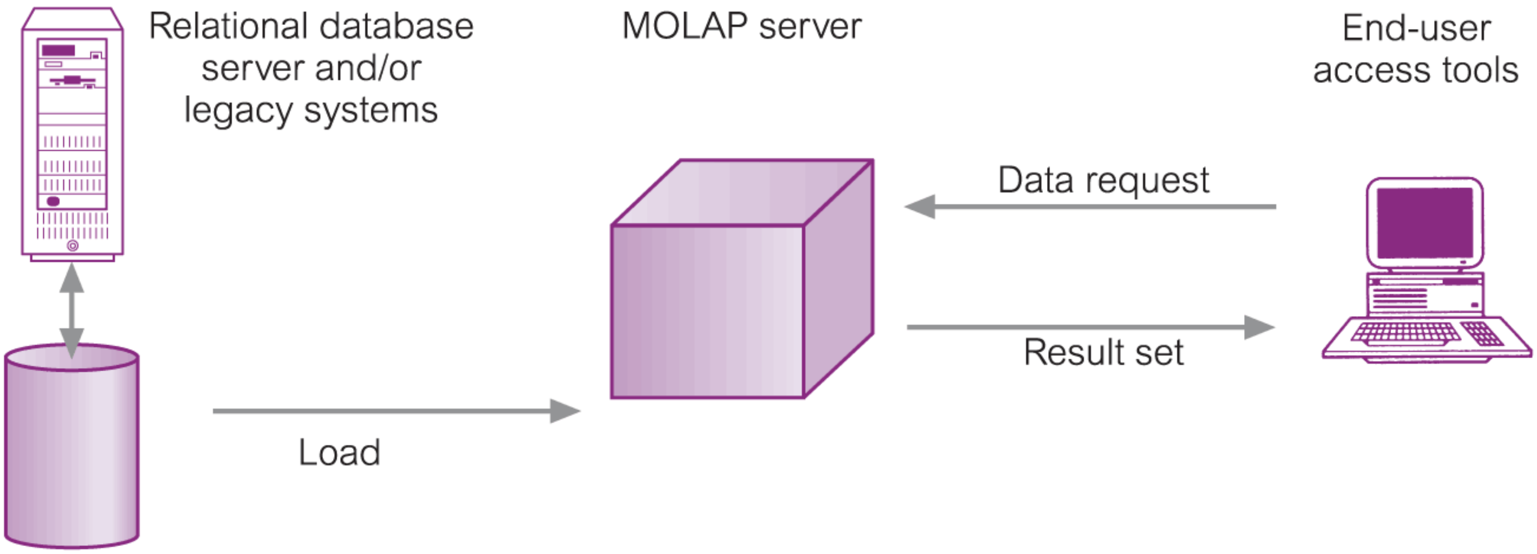
- Pre-aggregation is valuable, as typical dimensions are hierarchical in nature (e.g. Time dimension hierarchy - years, quarters, months, weeks, and days)

Categories of OLAP Tools

- **OLAP** (**On Line Analytical Processing**) tools are categorised according to the architecture used to store and process multi-dimensional data.

- Four main categories:
  - Multi-dimensional OLAP (MOLAP)
  - Relational OLAP (ROLAP)
  - Hybrid OLAP (HOLAP)
  - Desktop OLAP (DOLAP)

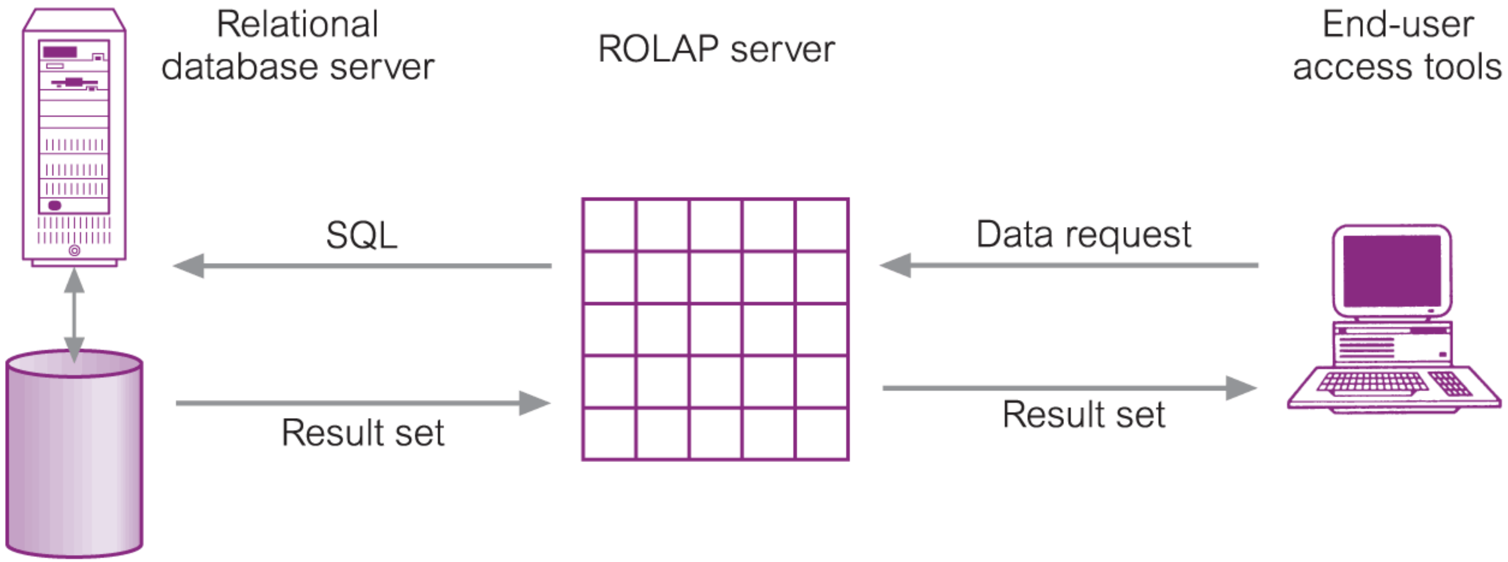
Typical Architecture for MOLAP Tools



Multi-dimensional OLAP (MOLAP)

- Use specialised data structures and multi-dimensional Database Management Systems (MDDBMSs)
  - Array technology and efficient storage techniques minimize disk space requirements
  - Provide excellent performance when data is used as designed, and focus is on data for a specific decision-support application.
  - Traditionally, require a tight coupling with the application layer and presentation layer.
  - Recent trends segregate the OLAP from the data structures through the use of published application programming interfaces (APIs).

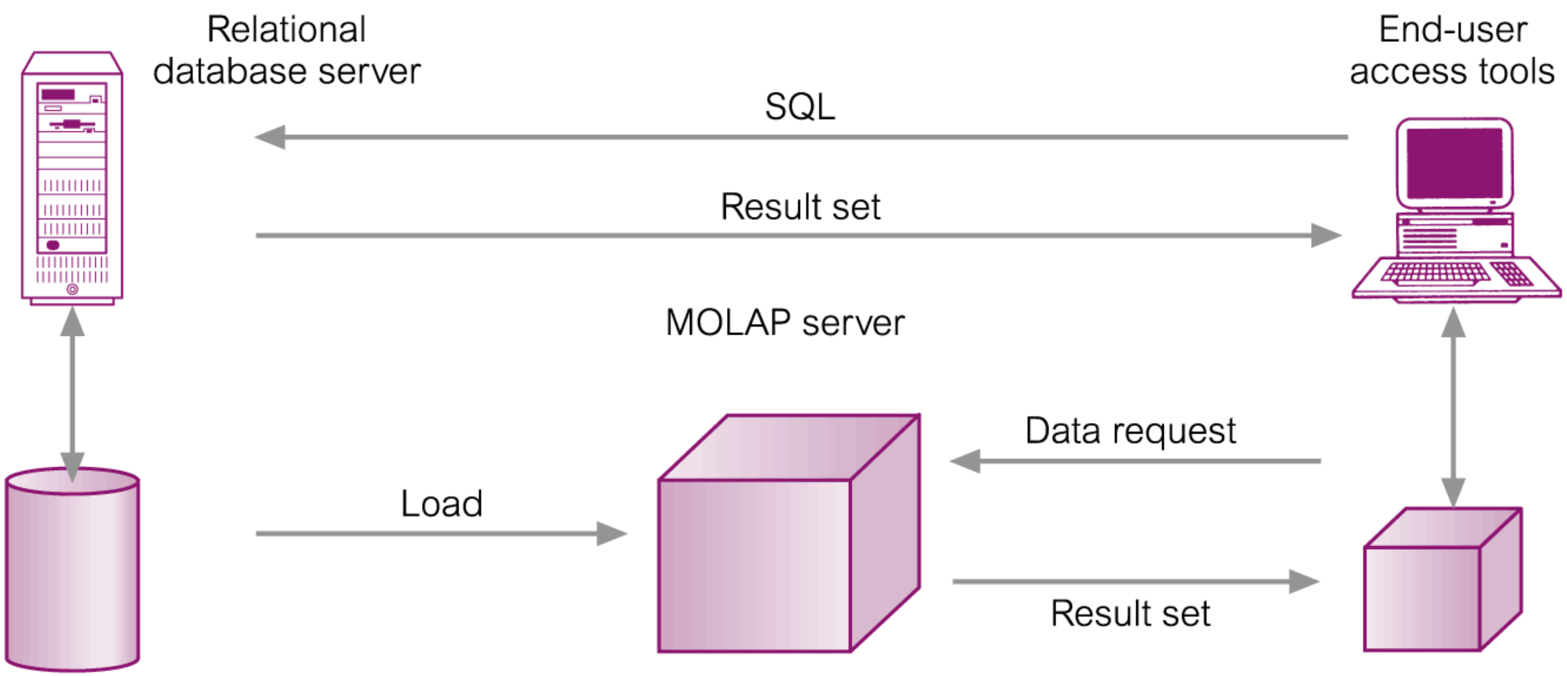
Typical Architecture for ROLAP Tools



Relational OLAP (ROLAP)

- Fastest-growing style of OLAP technology due to
  - need to analyse ever-increasing amounts of data
  - the realisation that users cannot store all the data they require in MOLAP databases.
- Supports RDBMS products using metadata layer
  - avoids need to create a static multi-dimensional data structure
  - facilitates the creation of multiple multi-dimensional views
- To improve performance, some products use SQL engines to support the complexity of multi-dimensional analysis
- Others recommend or require the use of highly denormalised database designs such as the star schema.

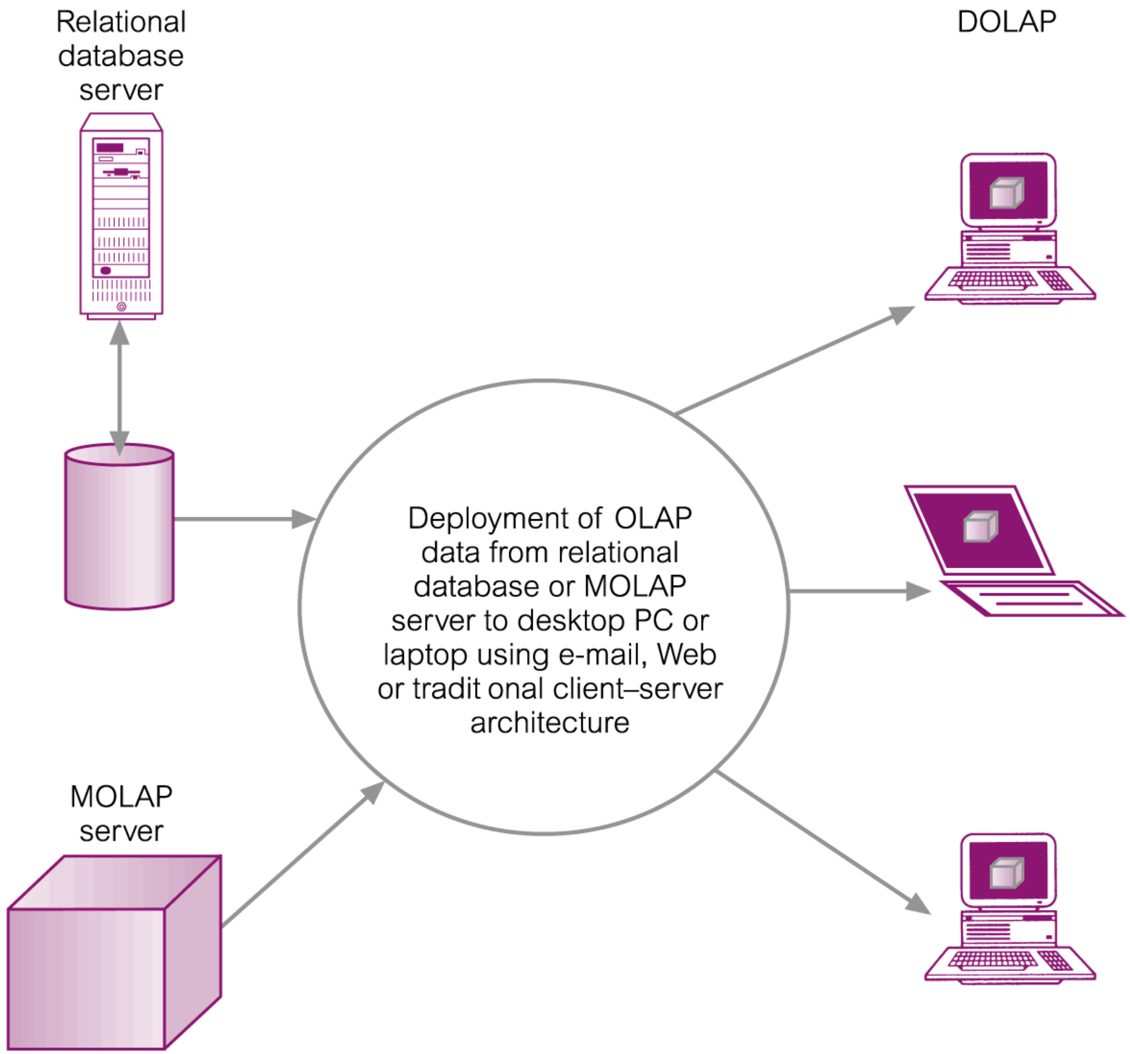
Typical Architecture for HOLAP Tools



Hybrid OLAP (HOLAP)

- Provide limited analysis capability
  - either **directly against RDBMS** products
  - or by using an **intermediate MOLAP** server.
- Deliver selected data directly from the DBMS or via a MOLAP server to the desktop (or local server) in the form of a datacube, where it is stored, analysed, and maintained locally.
- Promoted as being relatively simple to install and administer with reduced cost and maintenance.

Typical Architecture for DOLAP Tools

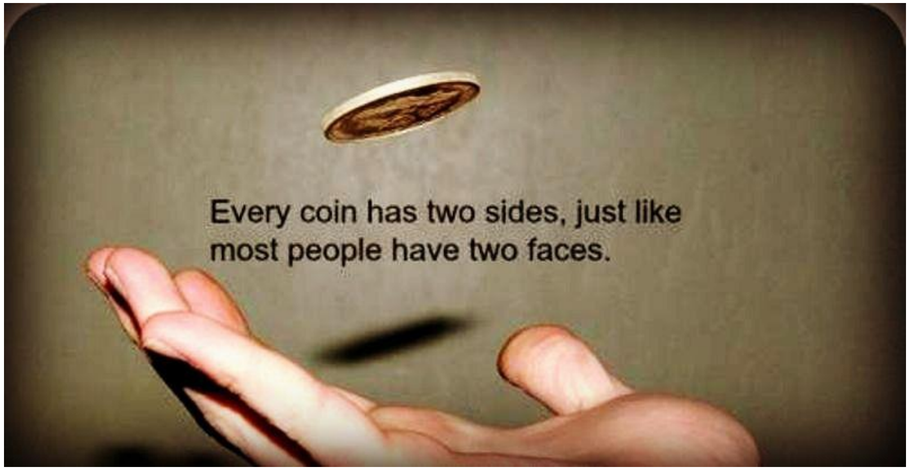


Desktop OLAP (DOLAP)

- Stores the OLAP data in client-based files and supports multi-dimensional processing **using a client multi-dimensional engine.**
- Requires that relatively small extracts of data are held on client machines.
  - May be distributed in advance, or created on demand (possibly through the Web).
- As with multi-dimensional databases on the server, OLAP data may be held on disk or in RAM
  - Some DOLAP products allow only read access.
- Most vendors of DOLAP exploit the power of desktop PC to perform some, if not most, multi-dimensional calculations.

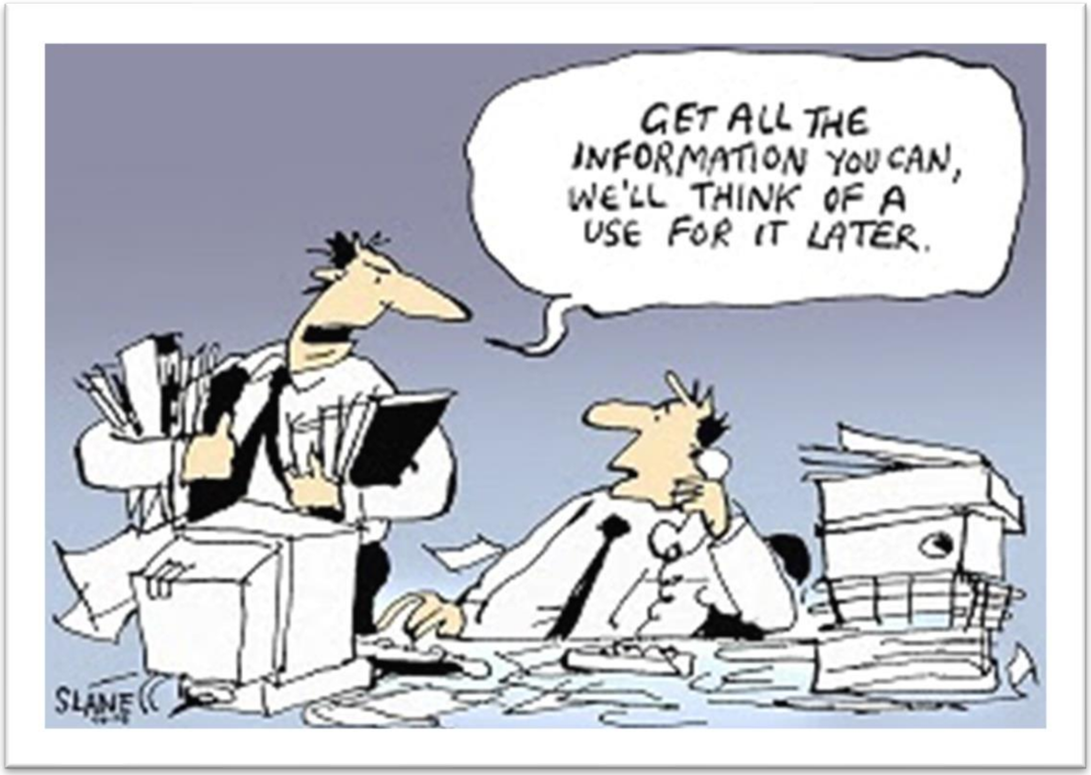
Limitations of dimensionality

1. The **multidimensional database can take up significantly more** computer storage room than a summarized relational database
2. Multidimensional products **cost significantly more** than standard relational products
3. Database loading **consumes significant system resources** and time, depending on data volume and the number of dimensions
4. **Interfaces and maintenance** are more complex in multidimensional databases than in relational databases



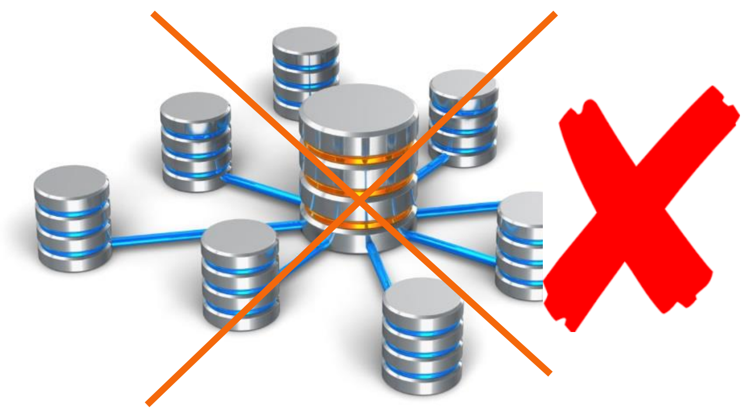
Pop Quiz

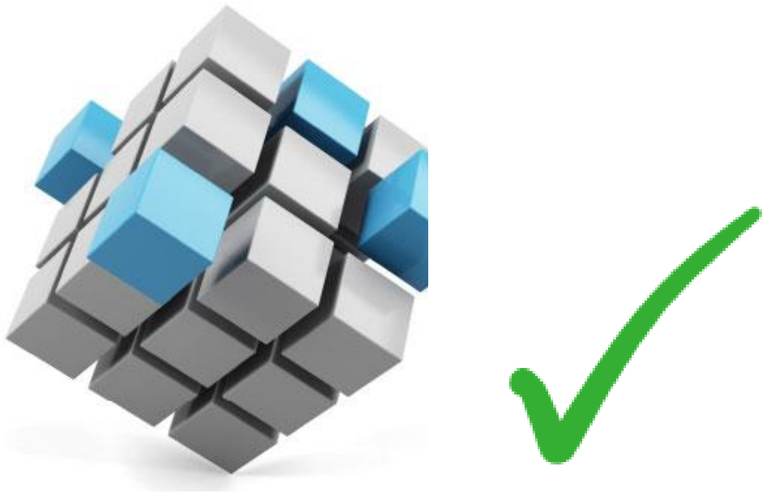
Is a warehouse of data a data warehouse?



Conclusion

A warehouse of data
is NOT a DW, it’s the ETL +
multidimensional OLAP Cube that
makes a DW!






Thanks!!!
Any question?

Research insights

International Journal of Business Intelligence Research
Volume 7 • Issue 2 • July-December 2016

An End-User Metadata Model on Object and Element Levels for Business Intelligence Users

ABSTRACT

The effective use of metadata can offer end users an improved understanding and greater level of assurance during the Business Intelligence (BI) report analysis process. This paper reports key findings from a case study that investigates critical end-user metadata issues in a large Australian organization. The findings led to the development of an end-user metadata model on object (report and cube) and element (term and column) levels, which can support effective BI use and potentially increase user satisfaction at the case organization. The adoption and use of BI applications by business stakeholders may be improved by incorporating the end-user metadata model.

KEYWORDS

Business Intelligence, Element, End User, Metadata, Object