INFO20003 Database Systems

Dr Renata Borovica-Gajic

Lecture 19

Data Warehousing

Semester 1 2018, Week 10

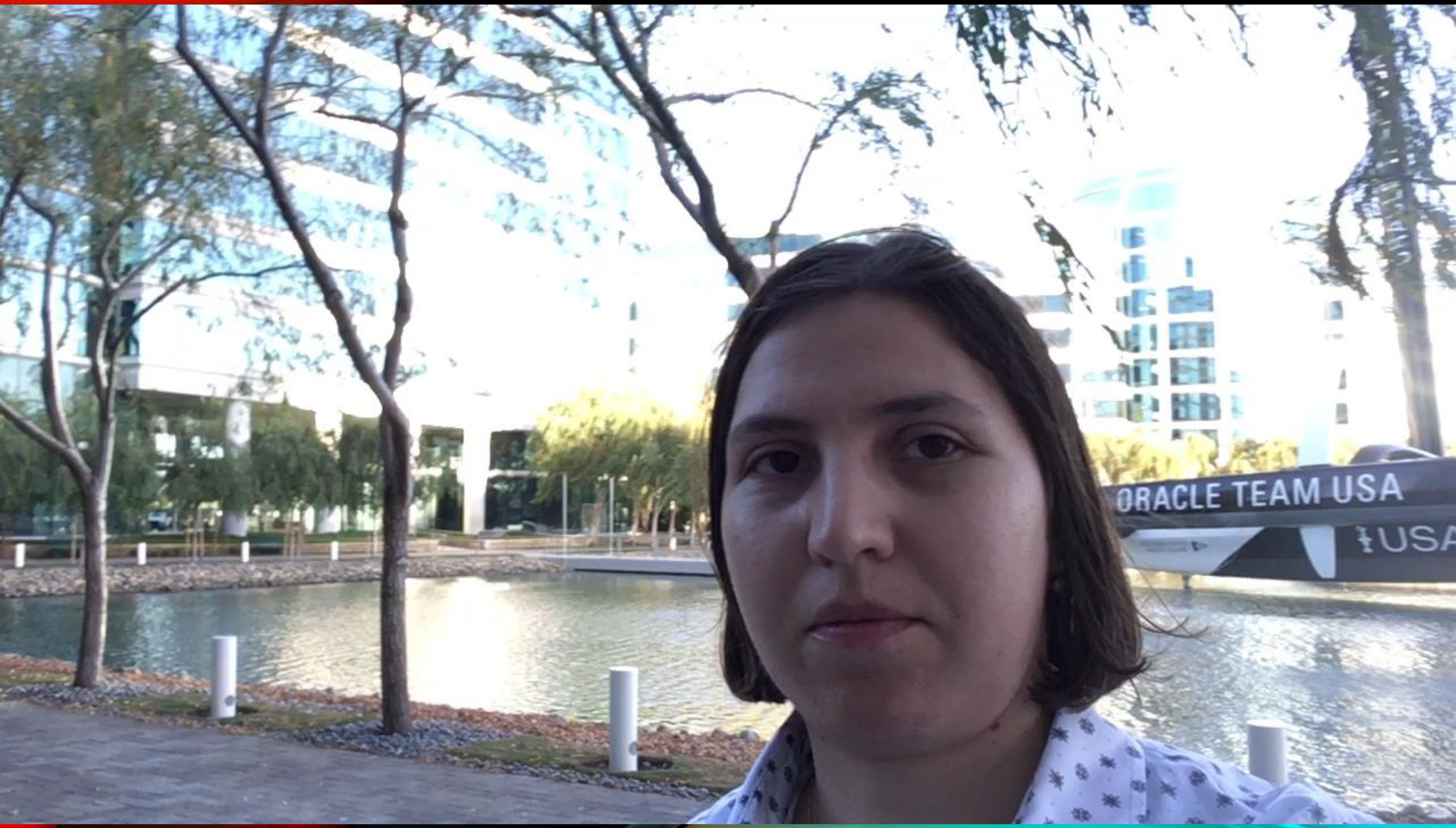
Slides adapted from Ida Asadi's work

WHY ARE DATABASES COOL ?

Featuring people from industry

Part 3

ORACLE LABS, USA



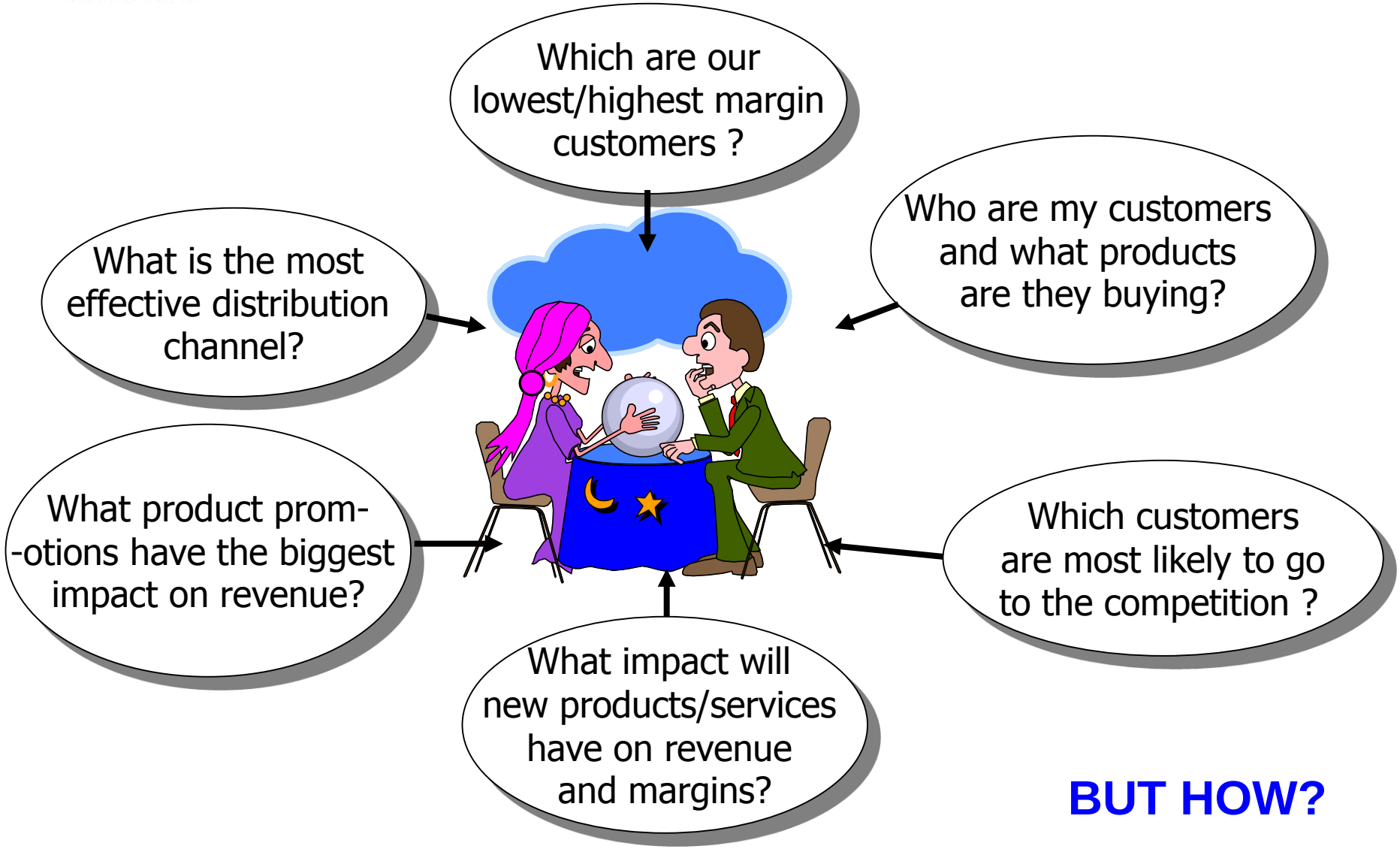
Coverage

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

- Articulate the differences between **transactional** (operational) and **informational** (dimensional) databases
- Explain the **characteristics of a DW**
- Understand and explain the **overall architecture of a DW**
- Design **Star Schemas**

Part 1: Introduction to Data Warehousing

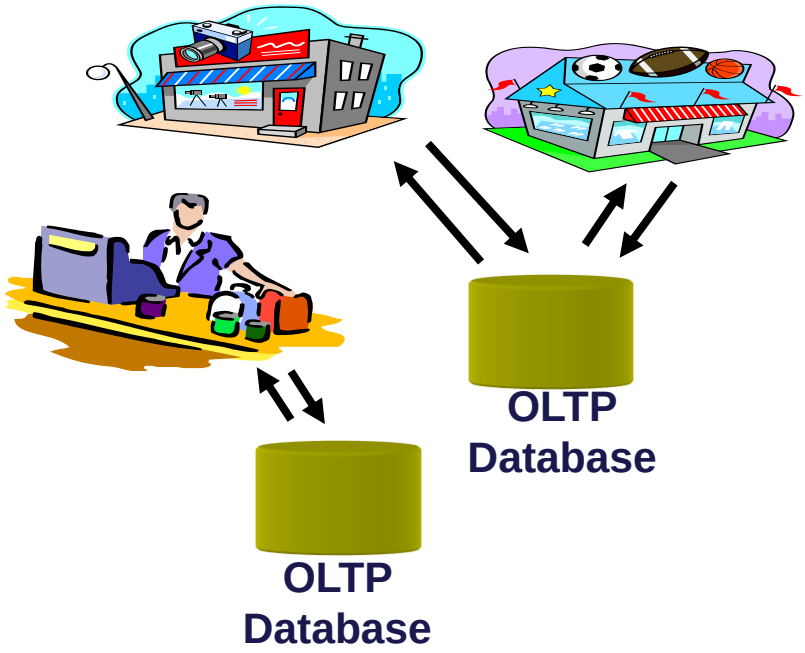
Motivations: A manager wants to know….



BUT HOW?

Relational Databases for Operational Processing

- Used to run day to day business operations
- Automation of routine business processes
  - Accounting
  - Inventory
  - Purchasing
  - Sales
- Created huge efficiencies



Databases are great, BUT for business…

- Too many of them
  - Everybody wanted one, or two, or more
  - Production, Marketing, Sales, Accounting …
- Everybody got what was best for them
  - IBM, Oracle, Access, Microsoft
- Eventually this re-created the problem databases were meant to solve
  - Duplicated data
  - Inaccessible data
  - Inconsistent data

But data is useful for analysis and decision making

What can we do about it?

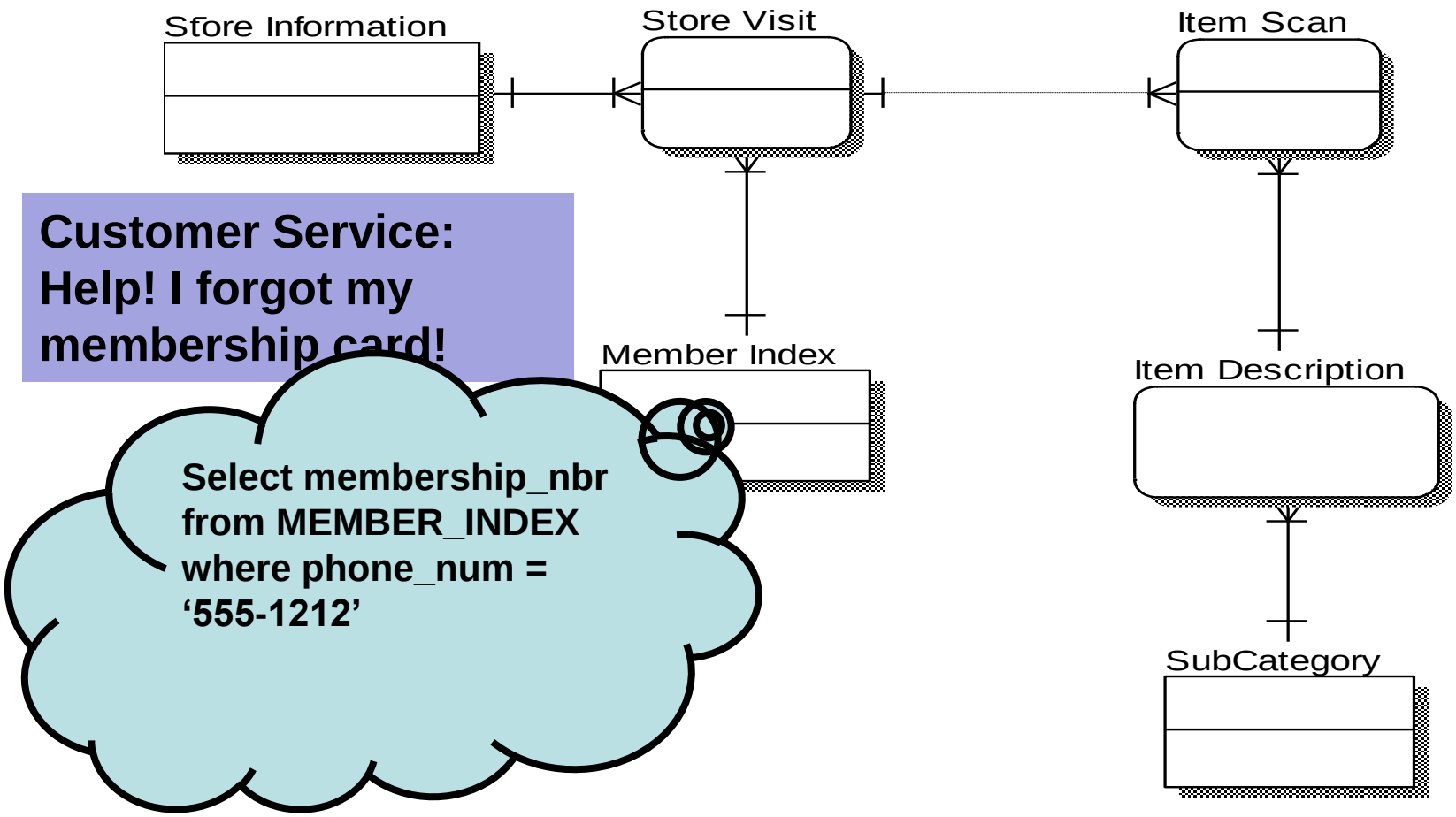
- Need an integrated way of getting the ENTIRE organisational data

- Its really an Informational Database, rather than a Transactional Database
  - A single database that allows *all* of the organisations data to be stored in a form that can be used to support organisational decision processes

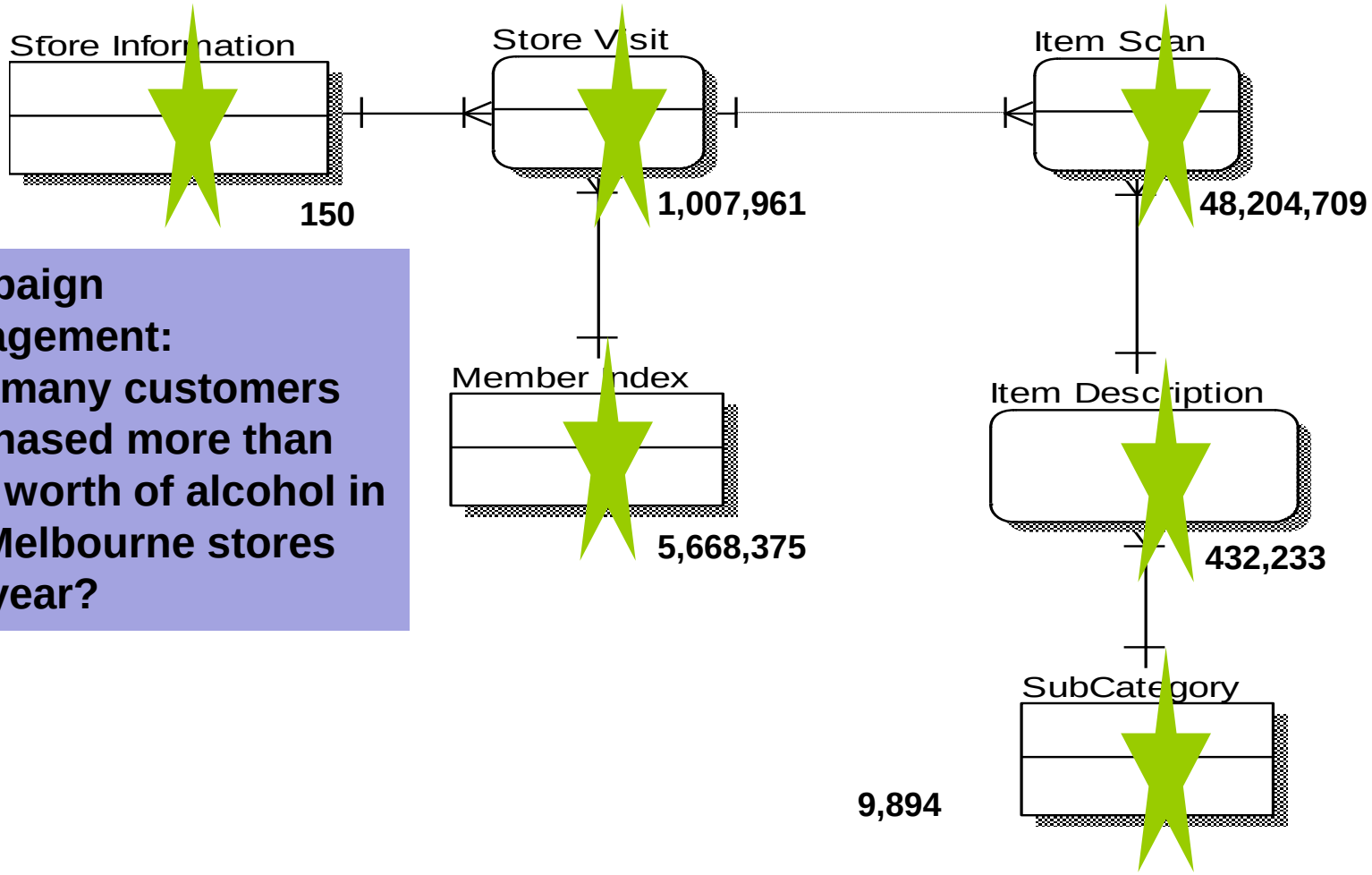
Warehouse: An Informational Database

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

Transactional (Operational) Questions



Analytical Questions



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

DW Supports Analytical Queries

- One is interested in numerical *aggregations*
  - How **many**?
  - What is the **average**?
  - What is the **total cost**?

- One is interested in understanding *dimensions*
  - Sales **by state by customer type**
  - Sales **by product by store by quarter**

DW will help answer these questions

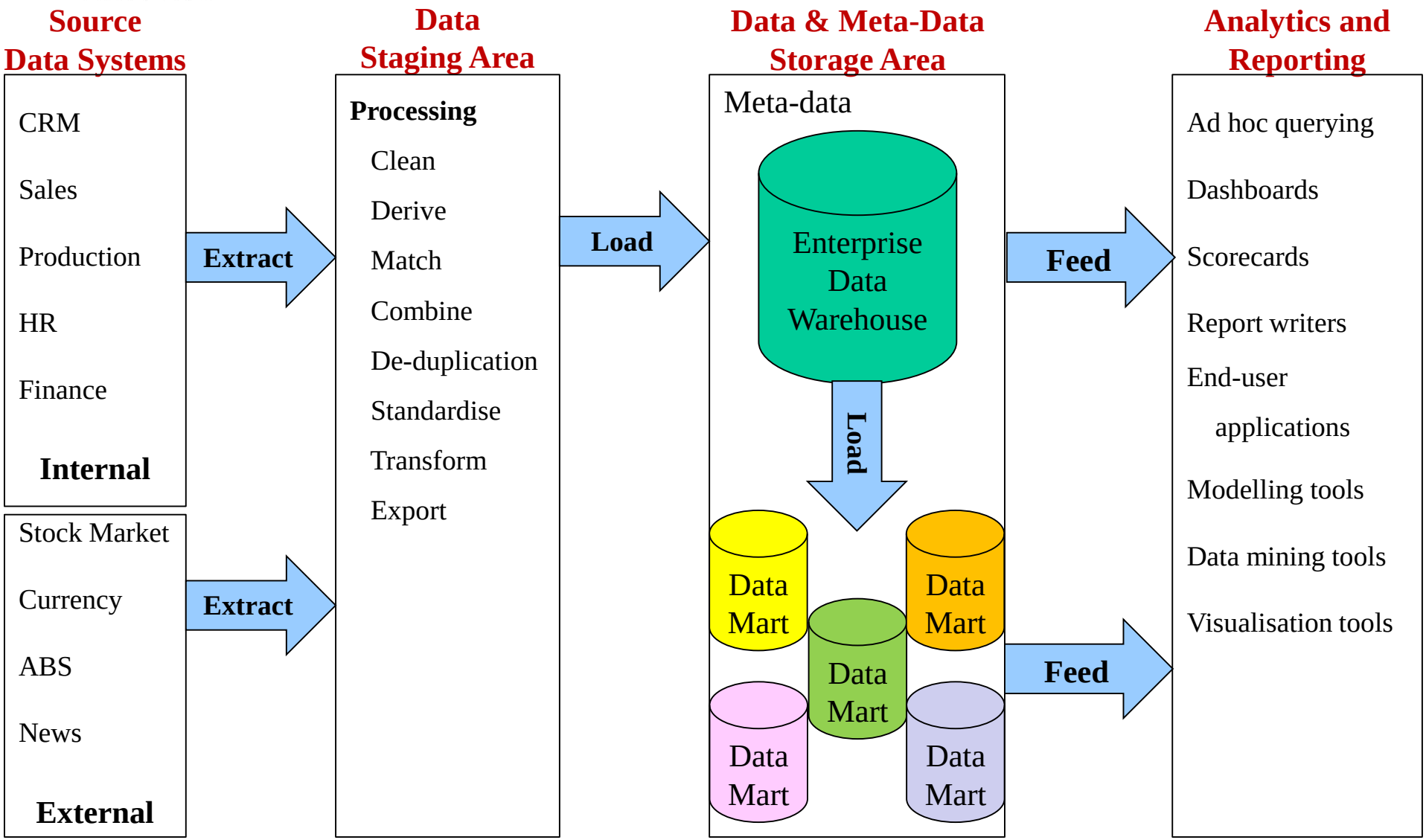
Characteristics of a DW

- Subject oriented
  - Data warehouses are organised around particular subjects (sales, customers, products)
- Validated, Integrated data
  - Data from different systems converted to a common format: allows comparison and consolidation of data from different sources
  - Data from various sources validated before storing it in a data warehouse

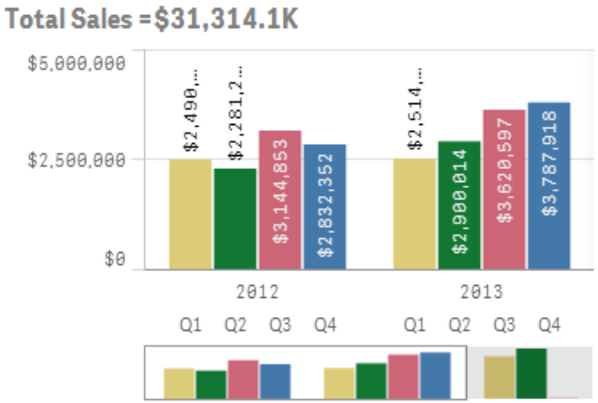
Characteristics of a DW

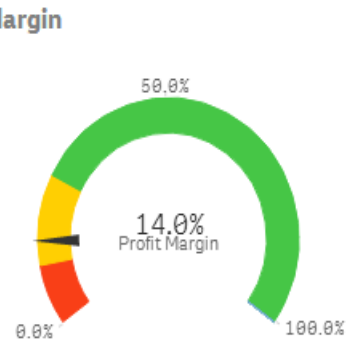
- Time variant
  - Historical data
  - Trend analysis crucial for decision support: requires historical data
  - Data consists of a series of “snapshots” which are time stamped
- Non-volatile
  - Users have read access only – all updating done automatically by ETL process and periodically by a DBA

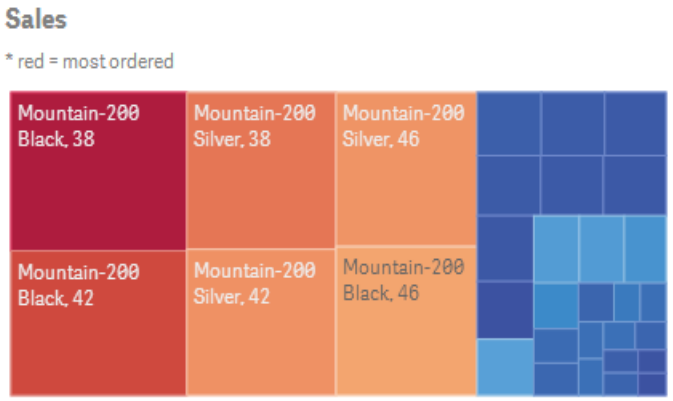
A DW Architecture

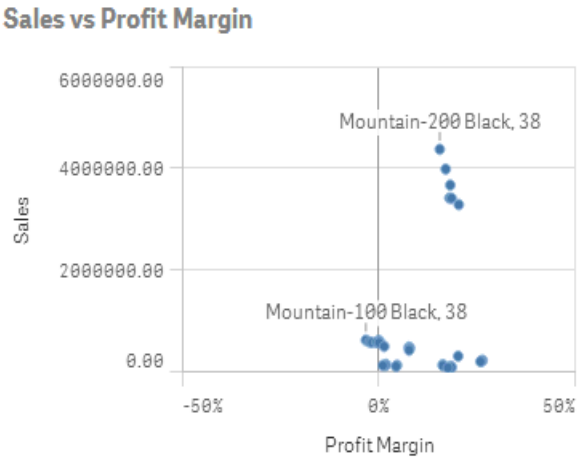


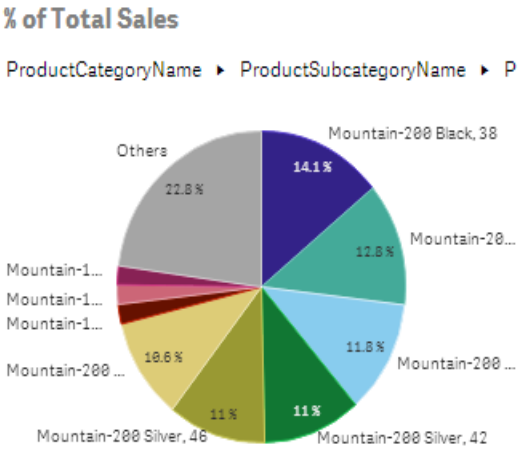
Business Intelligence Dashboard

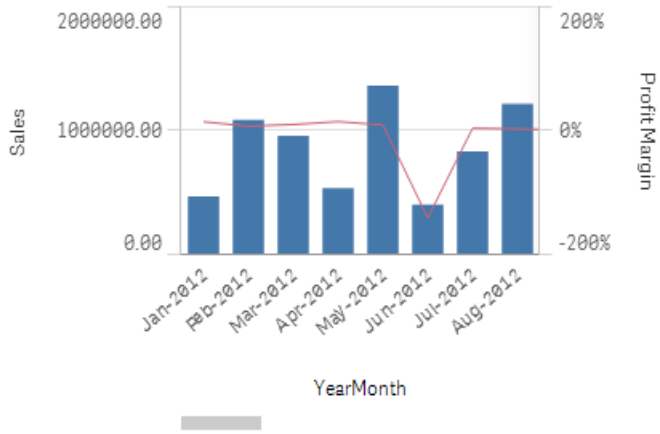






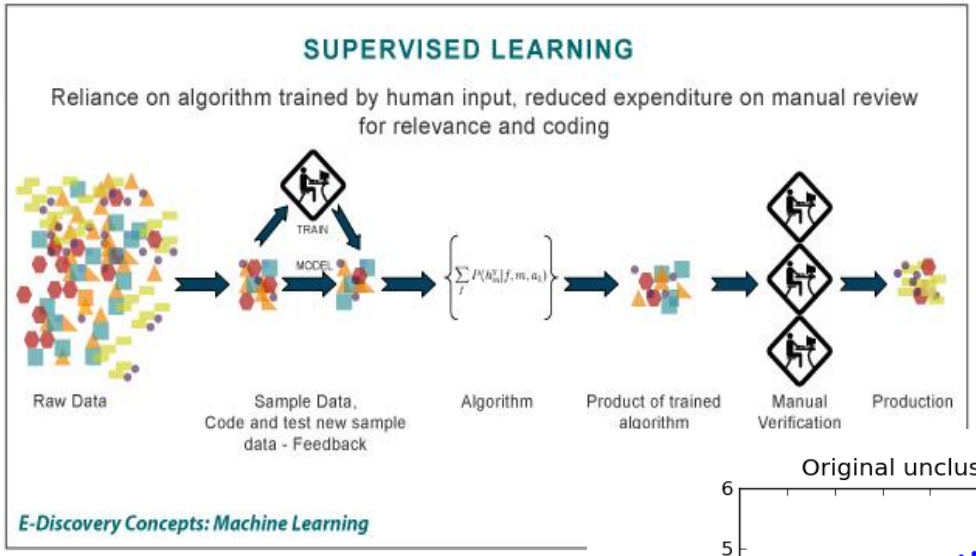


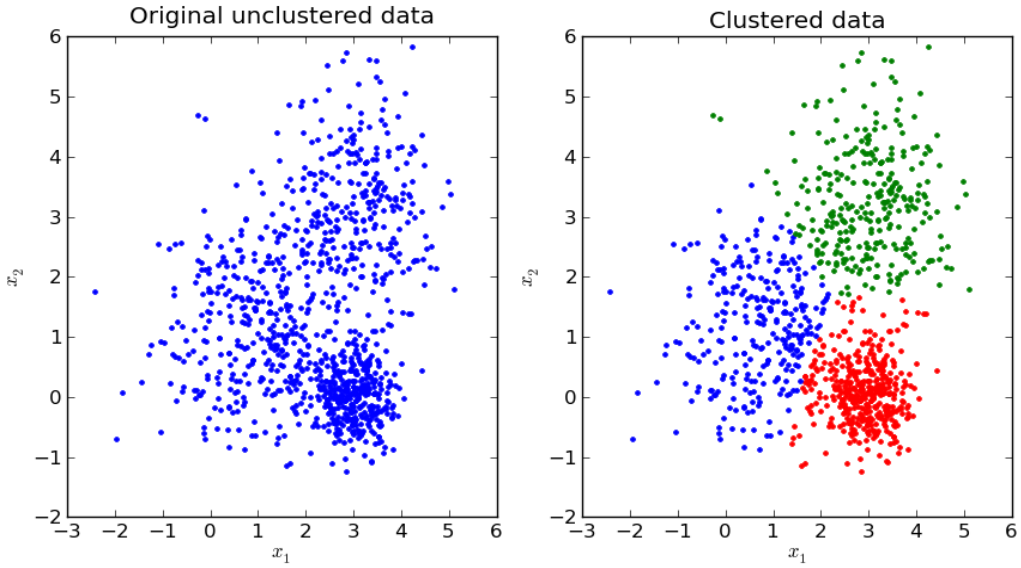




Qlik

DW Supports Advanced Analytics





http://us.hudson.com/legal/blog/postid/513/predictive-analytics-artificial-intelligence-science-fiction-e-discovery-truth
http://pypr.sourceforge.net/kmeans.html

Part 2- Dimensional Modelling

Business Analyst World

- How much **revenue** did the **product G** generate in the **last three months**, broken down by month for the south eastern sales **region**, by individual **stores**, broken down by **promotions**, compared to estimates and to the previous version of the product
  - Analysis starts usually with a single indication of something strange, then goes deep into the data, left to a new dimension, right to another, up to the summary, back down and left and right again, until the problem is identified…
  - Dimensional Analysis: To support business analysts view
    - Revenue per product per customer per location?

Fact

Dimension

Dimension

Fact Dimension Dimension Dimension

Introduction to Dimensional Modelling

- Popularised by Ralph Kimball in the 1990s
- Based on the *multi-dimensional* model of data and designed for retrieval-only databases
- Very simple, intuitive, and easily-understood structure
- Also known as *star schema* design

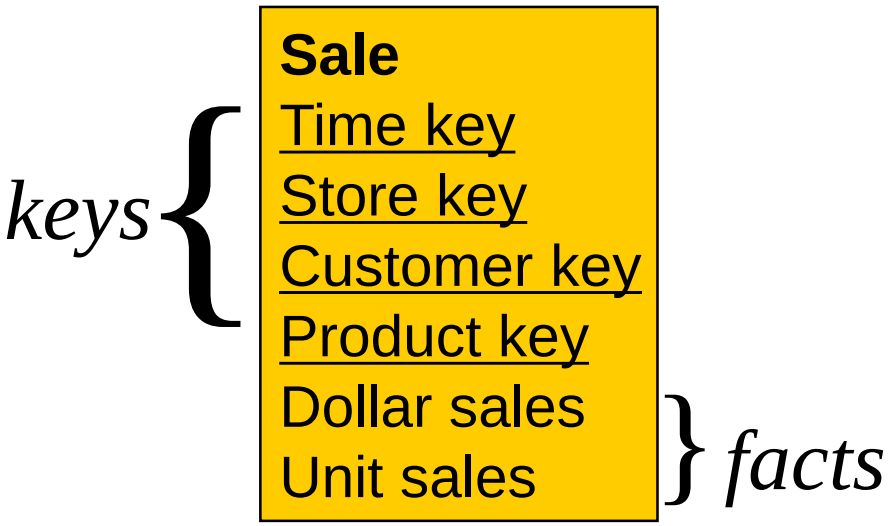
Dimensional Modelling

- A dimensional model consists of:
  - Fact table
  - Several dimensional tables
  - (Sometimes) hierarchies in the dimensions

- Essentially a simple and restricted type of ER model

Fact Table

- A fact table contains the actual business measures (additive, aggregates), called *facts*
- The fact table also contains *foreign keys* pointing to *dimensions*

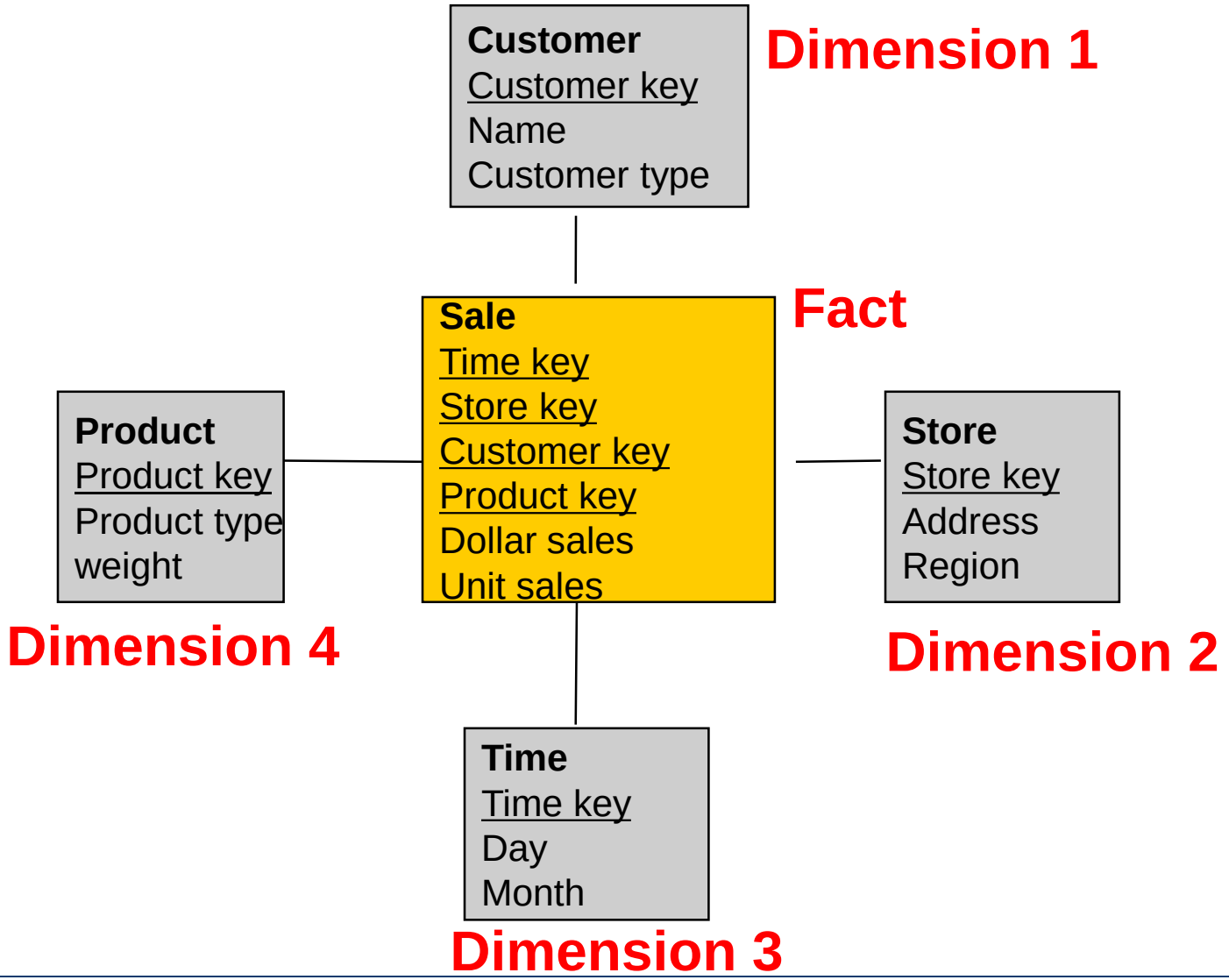


Fact Table - example

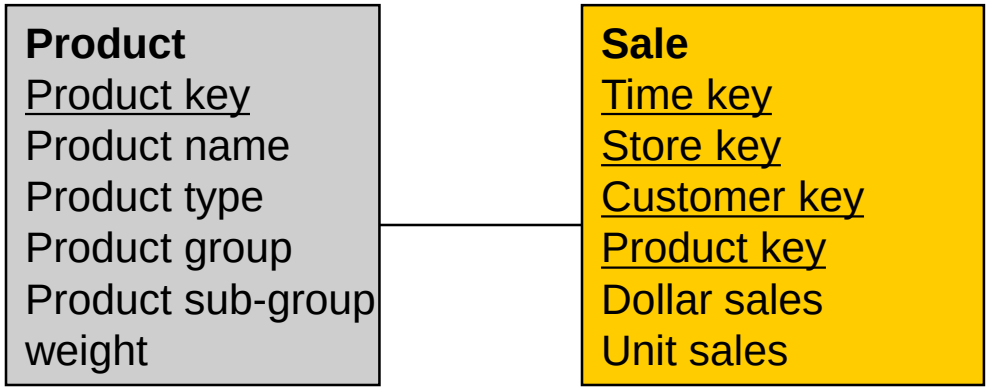
- Actual data might look like this
- Granularity, or level of detail, is a key issue
  - Finest level of detail for a fact table, determined by the finest level of each dimension

Time-id	*Store-id*	*Cust-id*	*Prod-id*	*Dollar sales*	*Unit Sales*
T100	S303	C101	P98	$120,000	5,000
T101	S303	C256	P98	$240000	10,000
T102	S387	C101	P10	$456,000	27,899
T100	S234	C400	P56	$100,200	5,600

Star schema – dimensional model



Dimension Hierarchies



Product name e.g. Hammer
- Product type e.g. Tool
  - Product group e.g. Hardware

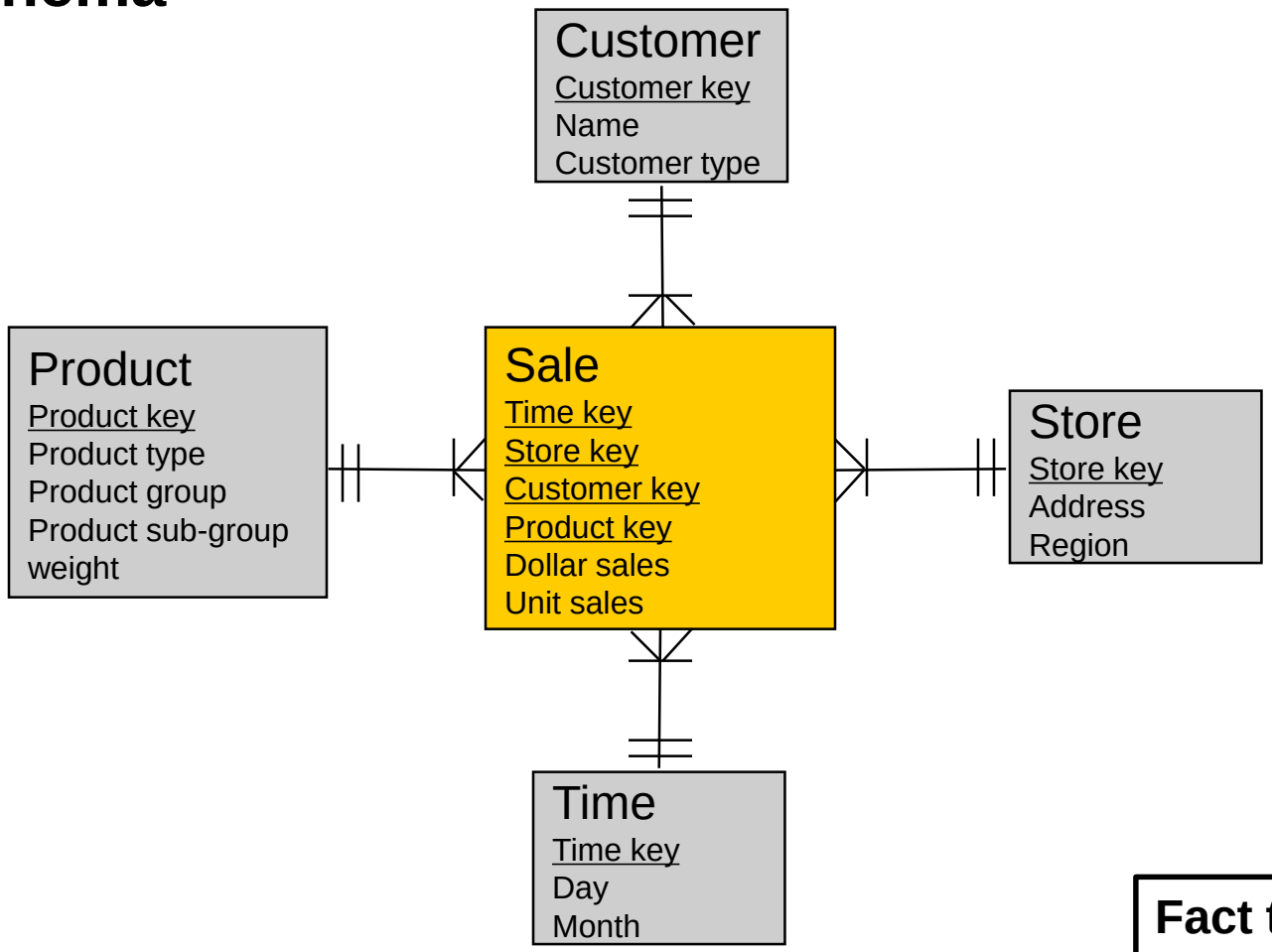
Dimension Table - example

- Actual data might look like this
- Hierarchy evident in data

Prod-id	*Prod-Name*	*Prod-Group*	*Prod-Subgroup*	*Weight*
P10	Hammer	Hardware	Tool	5kg
P56	10cm Nails	Hardware	Nails	1kg
P98	Plastic Pipe	Plumbing	Pipe	1kg

Dimensional model as an ER model

Star schema



Fact table is an intersection table

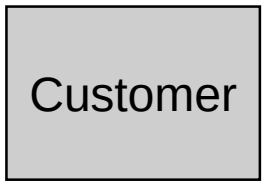
Designing a Dimensional Model

Steps:

1. Choose a Business Process
2. Choose the measured facts (usually numeric, additive quantities)
3. Choose the granularity of the fact table
4. Choose the dimensions
5. Complete the dimension tables

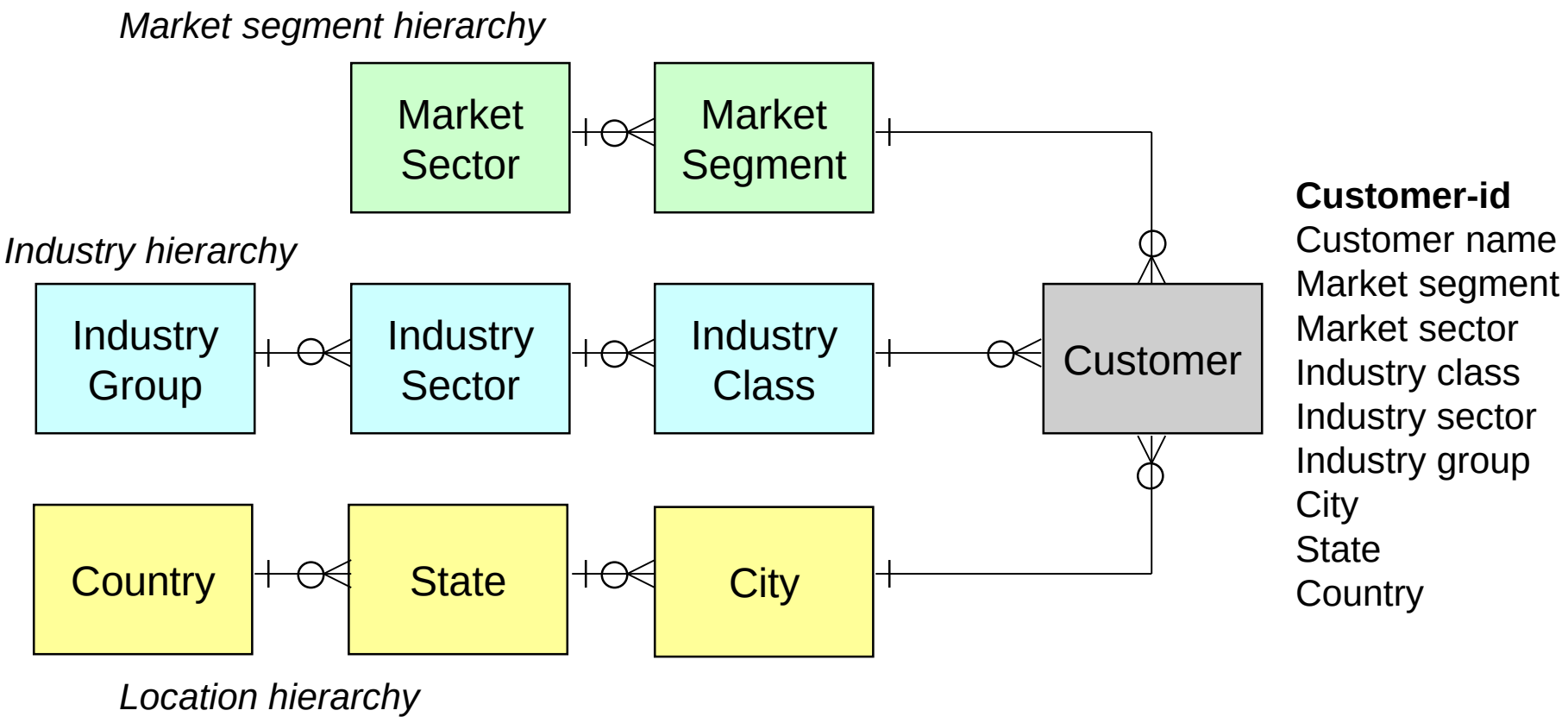
(Kimball, 1996)

Embedded Hierarchies in Dimensional Tables

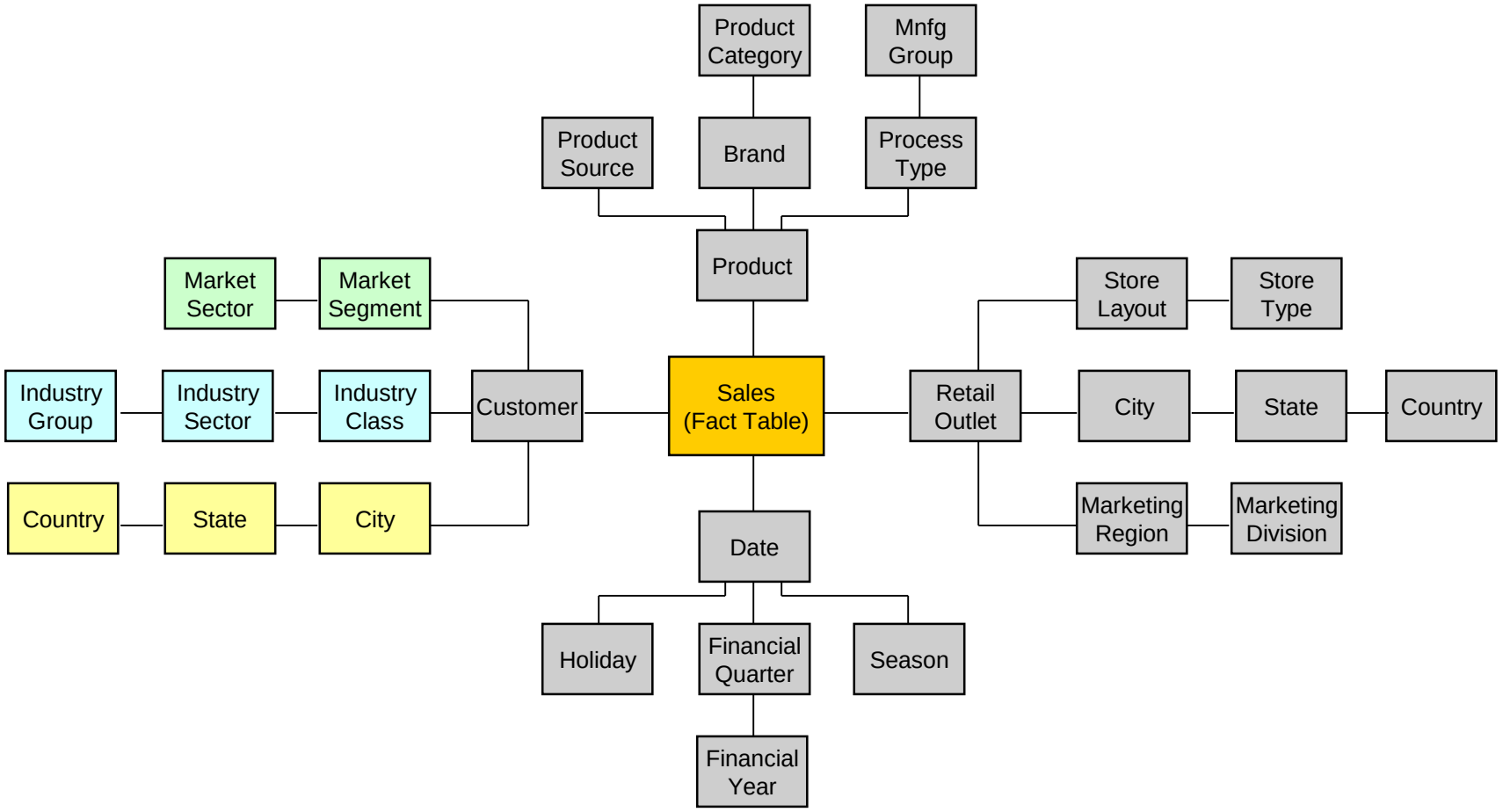


Customer-id
Customer name
Market segment
Market sector
Industry class
Industry sector
Industry group
City
State
Country

Embedded Hierarchies in Dimensional Tables

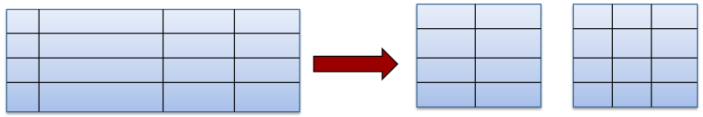


Snowflake Schema: hierarchy in dimensions

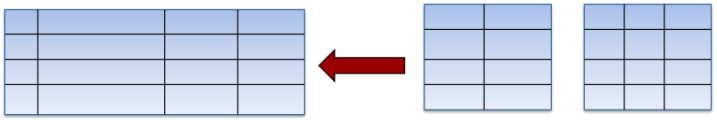


Design Outcomes: Normalised or Denormalised?

- Normalisation
  - Eliminates redundancy
  - Storage efficiency
  - Referential Integrity



- Denormalisation
  - Fewer tables (fewer joins)
  - Fast querying
  - Design is tuned for end-user analysis



Let's try it …

- We are making a data warehouse for a real estate agency. The company wants to track information about the **selling** of their properties. This warehouse keeps information about the **agents** (license#, first name, last name, phone #), **buyers** that come in (buyer id, first name, last name, phone #), and **property** (property#, property address, price). The information managers want to be able to find is **the number of times a property is viewed**, **sales price**. The information needs to be accessible **by rental agent, by buyer, by property** and **for different time** (day, week, month, quarter and year).
- Draw a star schema to support the design of this data warehouse.

What is Examinable?

- Differences between transactional and informational databases
- Designing a star schema
- Defining facts and dimension tables

Next Lecture

- Transactions