Database Systems

Tutorial Week 11

Objectives

I. Understand the fundamentals of dimensional modelling
II. Design a dimensional model using Kimball’s four-step design process
III. Discuss the impact of grain on fact tables

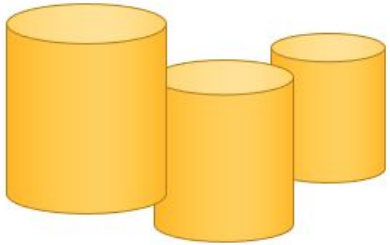
fully detailed (not aggregated)

designed for ***doing business***

Operational databases

read, insert and update

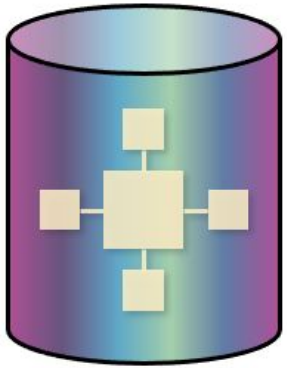
normalised



often specific to one department of a business

answer specific, low-level queries

"What's the total quantity of items in order 1295?"



Extract
Transform
Load

business-wide

facts organised by dimensions (non-normalised)

may be aggregated

Data warehousing

read and insert lots of data

answer high-level questions

designed for ***management, analysis and reporting***

"How much revenue did our new product range make? Break it down for me by product, location and time of day."

Key Concepts

- Data warehouse
  - Single database of *organisational data*
  - Supports high-level decision-making processes and data analysis
  - Allows all of the organisation's data to be stored in a form to support managers' decisions
  - Data is integrated from multiple sources (both internal and external to the organisation)
    - This is done by converting data into a common format and validating before storing it
      - This ensures credibility of the warehouse
  - Keeps historical data
  - ER model: data is organised around *conceptual entities*
  - Data warehouse: data is organised around *business processes* such as sales, finance or marketing

Key Concepts

- Business event
  - Event that occurs as part of a business process
    - For the sales business process, an individual order or sale = business event
    - For finance, a payment = business event
    - For a marketing data warehouse, view of a webpage or click on an online ad = business event
  - Data warehouses store info about *business events*

Key Concepts

- Dimension
  - An *entity* that describes and gives context to a business event
  - E.g. time, customers, products, locations
  - Suppose a CEO is interested in a comparison of revenue of a new model of the product with the older model in every quarter of the year by customer demographic group — what are the dimensions?
    - Time (“quarter of the year”), product (“product version”), customer (“customer demographic”)
  - Another example — for an insurance company where the key measurement (fact) is claims, the dimensions could be agent, policy, customer, time

Key Concepts

- Dimension table
  - Dimensions are represented in the data warehouse as dimension tables
  - Within each dimension table, you can store a range of attributes

Key Concepts

- Hierarchy
  - A sequence of attributes that describes a dimension across *different levels* of detail
  - E.g. for the dimension table “location” the data can be stored at various levels such as city, state, country etc.
  - For the dimension table “time”, you’ll have a hierarchy of day, week, month, quarter, year etc.
  - Hierarchies of dimensions are stored as *attributes* of the dimension tables
  - Hierarchies are used for selecting and aggregating data at a desired level of detail

Key Concepts

- Fact
  - A numeric measurement of a significant business event
  - Consider the scenario where a *customer* buys a *product* at a certain *location* at a certain *time*
    - The intersection of these four dimensions is a sale (the business event)
    - The sale can be measured in terms of the amount of revenue generated, number of items sold, total profit earned, etc.
      - These are all *facts* relating to the sale

Key Concepts

- Fact table
  - Facts are stored in a fact table
  - A row in a fact table corresponds to one or more business events
  - The intersection of dimensions that describe a business event
  - In general, the fact table has a composite PFK made up of the FKs connecting it to the dimension tables

Key Concepts

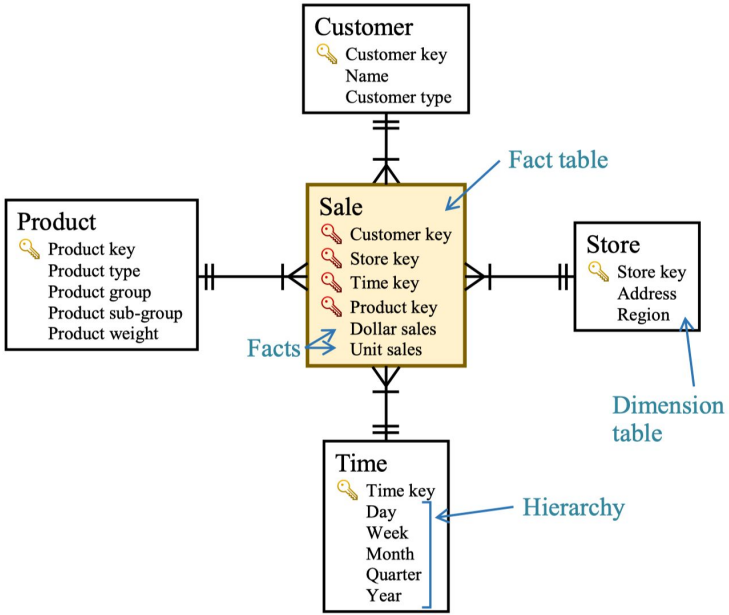
- Granularity / grain
  - The level of detail in a fact table
  - The fact table can store each business event in its own row (e.g. each sales event is an individual row)
  - Alternatively, it can store many business events aggregated together (e.g. sales data is aggregated down to one row per hour)
  - The finer the granularity is, the more precisely a query can extract details from the database

Key Concepts

- Dimensional model — the star schema
  - A model in which the fact table consisting of numeric measurements is related to all the dimension tables storing attributes
  - Fact table is at the centre
  - Dimensional tables are on the sides
  - This makes a *star schema*

Key Concepts

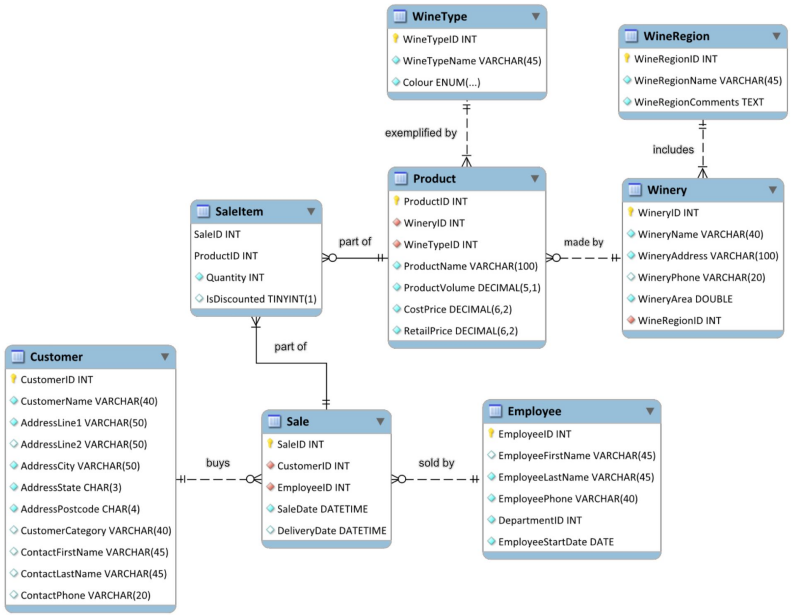
- This is a star schema with Sale as the fact table and Customer, Product, Store and Time as dimensions of the business



Exercise — Designing a Dimensional Model

Wimmera Wines is a large company that takes deliveries of grapes from wine growers, produces and bottles wine, and sells those bottles to retailers and restaurants. They produce many different types of wine at a range of price points, from cheap cask wine to top-of-the-range vintage bottles.

Wimmera Wines' day-to-day OLTP database uses the following ER model:



Exercise — Designing a Dimensional Model (5 mins)

The company is aiming to increase their product sales by 20% in comparison to the last 3 years. To help the business achieve their aim, you have been hired to design a data warehouse that can help business managers analyse data related to the sales theme.

The company is keen to understand all the aspects of their business that contribute to strong sales. For example, two business measures that have been mentioned are “total number of units of each product sold” and “revenue generated by each employee per year”.

a. As a class, brainstorm some more business measures that Wimmera Wines managers might need if they are to achieve their aim.

Exercise — Designing a Dimensional Model

a. As a class, brainstorm some more business measures that Wimmera Wines managers might need if they are to achieve their aim
Some examples include:

- Number of products sold per year
- Sales by a particular state
- Sales of a product in a given quarter of a year
- Revenue generated from a particular customer category
- Which product is selling the best (and hence generating the most revenue)

Exercise — Designing a Dimensional Model (15 mins)

b. Use Kimball's four-step dimensional design process to design a dimensional model for Wimmera Wines' product sales subject area.

i. Select and explain the business process.
ii. Declare the grain and justify your choice.
iii. Identify and explain the dimensions.
iv. Identify and explain the facts.

Exercise — Designing a Dimensional Model

b. Use Kimball's four-step dimensional design process to design a dimensional model for Wimmera Wines' product sales subject area

1. Select and explain the business process
    - Product sales
    - May use different measures associated with Sales to analyse them
2. Declare the grain and justify your choice
    - Wimmera Wines sells to retailers and restaurants, so they wouldn't make a large number of sales
        - Instead, each individual sale might include a large number of items
    - So it's appropriate to store *each* sale item as its own row in the fact table with no aggregation
    - But if we don't need such detailed information and weekly sales are sufficient, then we could aggregate the data by week

Exercise — Designing a Dimensional Model

b. Use Kimball's four-step dimensional design process to design a dimensional model for Wimmera Wines' product sales subject area

3. Identify and explain the dimensions
    - Looking at the ER model of the existing database and considering the business process (sales), we have these relevant dimensions:
        - Employee
        - Customer
        - Time
        - Product
4. Identify and explain the facts
    - We can extract these facts from the *source database*:
        - Unit sales
        - Dollar sales
        - Profit amount

Exercise — Designing a Dimensional Model

- Take a look at the final page of the solutions (uploaded on Canvas / GitHub) for a sample star schema for the Wimmera Wines data warehouse

Exercise — Fact Tables in Practice (7 mins)

Consider the following fact table:

Suppose the following sales data has been extracted from the business's operational database:

SaleID	SaleDate	CustomerID	CustomerCity	ProductID	Price	Quantity
54	2003-12-13 14:13	788	Melbourne	9644	$10.00	2
54	2003-12-13 14:13	788	Melbourne	8574	$15.00	1
67	2003-12-13 15:05	903	Melbourne	9644	$10.00	1
76	2003-12-13 17:26	322	Sydney	9644	$5.00	4
77	2003-12-14 09:58	292	Melbourne	8229	$15.00	2

a. Starting from this source data, how many rows will be inserted into the fact table if an hourly grain is selected?
b. How many rows will be inserted into the fact table if a daily grain is selected?
c. At which level of granularity can we answer questions about hourly sales? At which level of granularity can we answer questions about daily sales?

Exercise — Fact Tables in Practice

a. Starting from this source data, how many rows will be inserted into the fact table if an hourly grain is selected?
   - 5
   - None of these sale-item rows share the same hour, geography and product (keys of the Sale fact table)
   - No aggregation can be performed

b. How many rows will be inserted into the fact table if a daily grain is selected?
   - 4
   - Could aggregate first and third row into a single row in the fact table with DollarSales = $30.00 and Quantity = 3, since they share the same date, CustomerCity and ProductID
     - SaleID and CustomerID don't need to be the same since the keys of the Sale fact table are *time*, *geography* and *product*

c. At which level of granularity can we answer questions about hourly sales?
   - Grain must be hourly or finer

   At which level of granularity can we answer questions about daily sales?
   - Grain is daily or finer
   - E.g. if we have an hourly-grain fact table, up to 24 hourly rows can be aggregated (combined) into a single daily row when the fact table is queried, using a GROUP BY clause

Week 11 Lab

- Canvas → Modules → Week 11 → Lab → L11 Database Admin (PDF)
- Objectives:
  - Perform a logical backup
  - Import a table from an external data source
  - Do a back and simulate user error and partial logical recovery
- Breakout rooms, “ask for help” button if you need help or have any questions