Database Design in a Nutshell

Six steps of Database Design

Step 1: Requirements analysis. Collect information from customer about

- data;
- desired features of the database;
- information needs.

Step 2: Conceptual Database Design. Develop high-level description of data, describe constraints.

- High-level design: often done using Entity-Relatioship diagrams (E-R diagrams).

Step 3: Logical Database Design. Select a DBMS, convert high-level design into (relational) database design (*database schema*) in Data Definition Language (DDL) of the DBMS.

- DDL for relational databases is a part of SQL.

Steps 1–3 are main steps in database design. Three more steps, *enhance* the Logical design.

Step 4: Schema Refinement. Logical database design is analyzed and (potentially) improved.

- Goal of schema refinement: have database schema in one of **normal forms**.

Step 5: Physical Database Design. Tailor the database schema to expected workloads (queries, information needs).

- Choose *indexes*.
- *Tune* database design.

Step 6: Security Design. Identify user groups, information (parts of the database) to be made available to different user groups. Represent security information in DDL.

- SQL has some mechanisms to maintain security of the data.

Database Design vs. Application Design

Databases are rarely designed all by themselves. Typically, database design is accompanied by the database application design.

Important things to remember:

- Database design and database application design are **two different processes**. They may happen in parallel, but each has its own set of procedures to follow.
- Database design is studied in detail in this course. It concentrates on *determining the correct structure of the database* for a given application.
- Database application design is guided by the principles of software engineering. The design is broken into traditional stages:
  - Application requirements elicitation.
  - Application design.

  Database application design concentrates on *uses of data from the database* and on *properties of software* which would satisfy those uses.
- Database design and database application design are usually performed in parallel.

Requirements elicitation for database design

- Interaction with customers.
  - Who is the customer?
  - Who knows the structure of customer's data?
  - How can information about customer's data be obtained?
    - Interviews
    - Existing documentation
    - Specially prepared documentation
- Information collection.
  - How much domain expertise is needed?
  - What information is important? What information is NOT important?
  - How does one preserve elicited/discovered infromation?
    - Conceptual modeling.
    - Formalisms for conceptual modeling.
    - Entity-Relationship modeling framework.
    - Knowing the conceptual modeling mechanism that will be employed helps identify information to be elicited, the means of recording it.

Requirements elicitation for database application design

- Interaction with customers.
  - Who are the intended users of the application?
    - Possibly different categories of intended users with different *transaction needs* and *information needs*.
  - Who can provide *transaction needs* and *information needs* for the application?

    - Possibly, more than one source of information.
    - Possibly, source of informarion different from the database requirements expert.
  - How can information about customer's application needs be obtained?
    - Interviews
    - Existing documentation
    - Specially prepared documentation

- Information collection.
  - How does data get into the database?
  - How is data managed in the database?
  - What information needs exist?
  - How should retrieved information be displayed?
  - What are expected workloads?