Principles of Database Management: Solutions Manual

Wilfried Lemahieu

Dean and Professor at the Faculty of Economics and Business

KU Leuven

Bart Baesens

Professor at the Faculty of Economics and Business

KU Leuven

Lecturer at Southampton Business School

University of Southampton

Seppe vanden Broucke

Professor at the Faculty of Economics and Business

KU Leuven

Contents

Chapter 1. Fundamental Concepts of Database Management

Question 1.1 Discuss examples of database applications.

Answer

Some examples of database applications are:

- inventory: monitoring stock levels of products;
- point-of-sale (POS): storing which customer buys what item at what time and location (e.g., in a supermarket);
- geographical information system (GIS): stores and manages geographical data (e.g., Google maps);
- stock trading: used for deciding when to buy or sell stocks depending upon, e.g., macroeconomic trends;
- medical records: stores information about patients, their illnesses, and treatments offered;
- telematics: stores information about the driving behavior of a customer;
- human resources: stores information about employees, their departments, salaries, etc.
- video streaming: stores information about videos (e.g., YouTube).

Question 1.2 What are the key differences between the file versus database approach to data management?

Answer

In the file approach to data management, every application stores its data in its own dedicated files. Since each application uses its own data files, many using similar data, duplicate or redundant information will be stored, resulting in a waste of storage resources. Furthermore, there is a danger that data are updated in only one file and not elsewhere, yielding inconsistent data. In this file-based approach to data management, there is a strong coupling, or dependency, between the applications and the data. A structural change in a data file necessitates changes in all applications that use it, which is not desirable from a maintenance perspective. It is also hard to manage concurrency control (i.e., the simultaneous access by different users or applications to the same data without conflicts). Finally, since the applications each work independently, with their own ecosystem of data files, it is difficult and expensive to integrate applications aimed at providing cross-company services.

In the database approach to data management, all data are stored and managed centrally by a DBMS. The applications now directly interface with the DBMS instead of with their own files. The DBMS delivers the desired data upon request to each application. The DBMS stores and manages two types of data: raw data and metadata. Metadata refers to the data definitions that are now stored in the catalog of the DBMS. This is a key difference from the file-based approach. The metadata are no longer included in the applications, but are now properly managed by the DBMS itself. From an efficiency, consistency, and maintenance perspective, this approach is superior to the file-based approach. Another key advantage of the database approach are the facilities provided for data querying and retrieval. In the file-based approach, every application had to explicitly write its own query and access procedures.

To summarize, the file-based approach results in a strong application–data dependence, whereas the database approach allows for applications to be independent from the data and data definitions.

Question 1.3 Discuss the elements of a database system.

Answer

The key elements of a database system are as follows:

- Database model: provides the description of the database data at different levels of detail and specifies the various data items, their characteristics and relationships, constraints, storage details, etc. Also called database schema.
- Database state: represents the data in the database at a particular moment. Also called the current set of instances.
- Data model: provides a clear and unambiguous description of the data items, their relationships, and various data constraints from a particular perspective. Examples are the conceptual data model, logical data model, internal data model, and external data model.
- Three-layer architecture: an essential element of every database application; describes how the different underlying data models are related.
- Catalog: contains the data definitions, or metadata, of your database application. Also called the heart of the DBMS.
- Database users: users who interact with the database. Examples are information architect, database designer, database administrator, application developer, and the business user.
- DBMS languages: languages used for data definition or data manipulation. Examples are: data definition language (DDL) which is used by the database administrator to express the

database's external, logical, and internal data models; and data manipulation language (DML), which is used to retrieve, insert, delete, and modify data.

Question 1.4 What are the advantages of database systems and database management?

Answer

Databases, if adequately designed and managed, offer various advantages:

- data independence: refers to the fact that changes in data definitions have minimal to no impact on the applications using the data;
- database modeling: allows provision of an explicit representation of the data items together with their characteristics and relationships;
- managing structured, semi-structured, and unstructured data;
- managing data redundancy: redundant data can now be successfully managed;
- specifying integrity rules: data integrity rules can be explicitly defined that can be used to enforce the correctness of the data;
- concurrency control: a DBMS has built-in facilities to support concurrent or parallel execution of database programs, which allows for good performance;
- backup and recovery facilities: can be used to deal with the effect of loss of data due to hardware or network errors, or bugs in system or application software;
- data security: data security can be directly enforced by the DBMS;
- performance utilities: DBMSs come with various types of utilities aimed at improving response time, throughput rate, or space utilization.

CASES AND QUESTIONS

In this appendix, we give examples of questions that connect concepts across chapters. They can be used as exam or assignment questions. The questions can also easily be adjusted, extended, and/or combined.

Question 1

Fitness company "Conan" wants to set up a database for its members and trainers. One aim is to record information about which members participated in which sessions and which trainers supervised which sessions.

Conan operates various fitness centers in various cities. Every center is characterized by a unique name (e.g., Fitplaza, my6pack, etc.). Every center has an address and one or more rooms (you can consider the address as atomic). Every room has a maximum capacity. Within a center, each room has a unique number, such as 1, 2, 3, etc.

Customers can register for individual or group sessions in different centers. Each group session requires exactly one trainer. Individual sessions are done without a trainer. For each customer, we want to store the first name, family name, and date of birth. You can assume that the combination of first name, family name, and date of birth is unique. For each trainer, the diploma is also recorded. A person can be trainer in one session and participant in another session (either individual or group session). The data model should also include information about people (e.g., prospects) that have not participated in any sessions yet, or trainers (e.g., interns) who have not supervised any group sessions yet.

For each session, the date and starting hour should be recorded. For group sessions, the type of class should be stored also (e.g., aerobics, body styling, etc.). Sessions can start at the same time on the same day, but in different rooms of a center or in different centers. At a given start hour of a given day, at most one individual or group session can start in a given room of a given center.

- Make an EER conceptual data model for the data requirements. Discuss any semantics that cannot be modeled.
- Map the EER conceptual model to a logical relational model. Discuss any semantics that get lost in the mapping, or semantics that can be added.
- Give two examples of views in the external data model.
- Give two examples of SQL queries, one example of a trigger, and one example of a stored procedure.
- Give recommendations for the design of the internal data model. What primary and secondary file organization methods would you advise?
- Illustrate how two extensions provided by ORDBMSs could be useful.

- Discuss the usefulness of XML in storing information.
- Develop a corresponding data warehouse design.
- Give two examples of OLAP operations.
- Illustrate two applications of descriptive and predictive analytics.
- What data management roles would you suggesting hiring?

Question 2

Recently, the European Union provided funds to set up a cross-national research database that stores information about (peer-reviewed) scientific articles of researchers working at institutions in the EU. Science Connect is the company that will be setting up this database.

The system will store information regarding scientific staff (persons) and research institutions. Both are uniquely identified by a person ID and an institution code, respectively. Additionally, the following is also recorded for each person: phone number, keywords that identify his or her key research topics, and the institution for which he or she works. A person can be an author of one article and a reviewer of another peer-reviewed article at the same time.

The database will store the following information about scientific articles. Each scientific article is uniquely identified by a DOI (a Document Object Identifier), and the system also stores the title and the authors of the article. In the case of multiple authors, the position of each author is stored. Science Connect distinguishes between two types of scientific articles: a scientific article is either a peer-reviewed paper or a technical report. The system stores the citation count of peer-reviewed papers and which persons reviewed the paper. A technical report is always published by a single research institution, while research institutions can publish multiple technical reports.

The system also keeps track of the different scientific publishers (e.g., IEEE, Elsevier). A publisher is identified by its name. A publisher can publish multiple journals to which research institutions can subscribe. These journals are given a name by the publisher (e.g., *Decision Support Systems*). Publishers can have journals with the same name as other publishers. For example, it is possible that both IEEE and Elsevier have a journal with the title *Management Science*. The impact factor, which measures the scientific impact of a journal, is also stored.

Finally, peer-reviewed papers are published in journals; technical reports are published by a single research institution.

- Make a UML conceptual data model for the data requirements. Discuss any semantics that cannot be modeled.
- Map the UML conceptual model to an ODMG model. Discuss any semantics that get lost in the mapping, or semantics that can be added.
- Give two examples of OQL queries.
- Give recommendations for the design of the internal data model.
- Discuss two applications of descriptive and predictive analytics.
- What data management roles would you suggest hiring?

Question 3

One of your (hipster) acquaintances thinks he has the next billion-dollar startup idea for an app: Pizza Delivery with Entertainment. He explains the basic functionality of the app as follows: