DATABASE DESIGN I - 1DL300 – Winter 2015

Assignment 2

Data quality, normalization and re-design

Handling data is a necessity for most organizations since when they are founded. As a consequence, most existing organizations already have a database, and when you join one you will rarely asked to design a new database from scratch. However, it can be necessary to improve the existing database, which can suffer from an initial poor design or from poor maintenance. Therefore, it is important to be able to assess the quality of the data, the robustness of the schema to the insertion of inconsistent data, and the ability to include new information. In summary, it may be necessary to re-design and improve the existing database keeping the original data.

The purpose of Assignment 2 is to get an understanding of the different normal forms and of the problems that can be prevented by normalization. Furthermore, you have the opportunity to practice the process of extending a relational database given a specific scenario.

After completing the assignment, you should be able:

1) To determine if a database table conforms to the 1NF, 2NF, 3NF or the Boyce-Codd normal form (BCNF), and re-design them, i.e. by normalization.
2) To model real world scenarios in terms of entity-relationship (ER) models and to translate an ER diagram into a corresponding relational database implementation.
3) To write a technical report describing your activities, understandable by people without a background on RDBMSs.

Scenario

The Music Department of Uppsala University currently stores information about its musical scores on a single table, kept in an excel file. You find a portion of it on the student portal (archive.xls), where only a small subset of the original records has been kept for simplicity.

As the number of items in the department's library has been significantly growing during the last few years, you have been hired to store the existing data into a relational database. However, while doing so you have noticed several quality issues with the existing database, and you have decided to invest some time to redesign and improve it. To justify the number of hours devoted to this project, you must also prepare a report for the head of the Music Department, Prof. Stefano La Carpa, explaining him the problems you have identified and the solutions you have implemented.

Tasks

1) Before storing the data into a relational database, open the excel file and perform a data quality check. In particular:
    - o Understand the meaning of the different columns. (Please, check Appendix A for the coding of column INSTRUMENTS.)
    - o Identify missing values and replace them with empty fields, so that they are represented in a uniform way.
    - o Check if the data contained in the different columns complies with the semantics of that column, and if not move data from one column to the other.
    - o Standardize the INSTRUMENTS column in row 25 ("Strings and cembalo") following the format explained in Appendix A.

    NOTE 1: Save your updated file in CSV format.

    NOTE 2.3: remember to document all your actions, so that you can later explain them in your report.
2) Create a table to store the CSV file into a relational DBMS. You can either create a database on your laptop, or use MySQL Workbench and connect to the database server running on groucho.it.uu.se, port 3306. A database for your group has been created on the server, with name vt15_db<group_number>, username vt15_user<group_number> and password pwd<group_number>.
3) Import the data into the new table using the LOAD DATA SQL command.
4) Check if the table is in 1NF. If not, normalize it to 1NF. At the end of this process PRIMARY KEYs must have also been defined.
5) Use UPDATE SQL commands to populate the new columns.

    NOTE 1: to do this, you may need a sub-string function to extract parts of the strings in existing fields.

    NOTE 2: for some columns this operation can be difficult to automate and need manual INSERTs. You are not requested to perform these transformations, but you should still document them in the report.
6) State in which normal form (2NF, 3NF or BCNF) each of the tables in your current database is, and why.
7) For each table that does not fulfill the requirements for BCNF, explain the problems that this lack of normalization has and their potential consequences.
8) Redesign (decompose) the table(s) that do not fulfill the requirements for BCNF.
9) Create the new tables in the database on the MySQL server and populate them with all needed data. The data population has to be done using SQL queries retrieving the data from the old tables and inserting them into the new designed tables. Add primary and any foreign keys to your table definitions.
10) Produce an EER diagram corresponding to the final database.

Examination

You are asked to submit two files: a **PDF report** and a **text file with all the SQL commands**. Your report must be targeted to Prof. La Carpa and the IT personnel of the Music department. Prof. La Carpa is not an expert of relational DBMSs, so you should explain in technically accurate but understandable terms why the redesign of the database is necessary. Remember that he is going to pay for your work, so you must justify your redesign. The report will also be used by the IT personnel, so it must also contain all the technical details. In particular, your report must also include the following:

1) The (E)ER diagram of the re-designed database.
2) All full functional dependencies for the old (original) and new (created by you) tables, together with explanations why the tables, both old and new ones, are in their respective normal forms.
3) Descriptions of problems caused by the lack of normalization of a table, and the potential consequences.

As said, you should also submit a text file including all your SQL commands.

Appendix A: Format to specify instruments in orchestral works

An orchestral piece can involve many instruments. To quickly indicate all the instruments (and players) needed to perform a given piece, a shorthand notation can be used instead of listing all the instruments with their full names. The instruments are divided into 5 groups:

1) Flutes, oboes, clarinets, bassoons (Woodwinds)
2) Horns, trumpets, trombones, tubas (Brass)
3) Timpani, percussion, harp
4) Other instruments (piano, harpsichord, …)
5) I violins, II violins, violas, celli, basses (Strings)

Apart from Group 4, where it is not possible to list all instruments a priori, each instrument type can be specified as a number indicating how many instruments of that type are playing, in the same order of the list above. As an example:

2222.4331.121 piano (picc., engh) str (16 14 12 10 8)

Means that the piece is for:

1) 2 flutes, 2 oboes, 2 clarinets, 2 bassoons
2) 4 horns, 3 trumpets, 3 trombones, 1 tuba
3) Timpani, 2 percussionists, one harp
4) Piano, piccolo, English horn
5) 16 I violins, 14 II violins, 12 violas, 10 celli, 8 basses

Notice that in the "Other instruments" group some of them are between parentheses: this means that they are *doubled* instruments, played by the same people already listed in the previous groups – for example, the piccolo will be played by one of the two flute players. The alternative orchestration:

2222.4331.121 piano, picc. (engh) str (10 10 8 8 6)

indicates that the piccolo (picc.) is played by an additional musician. The number of players for each type of strings is sometimes omitted, in which case we can write:

2222.4331.121 piano, picc. (engh) str

and we mean the same as above.

This notation is only used for orchestral works, and not for small chamber music ensembles. Therefore, a piece for violin and piano would not be indicated as:

0000.0000.000 piano str (1 0 0 0 0)

but directly as: **Violin and piano**