NEGATIVE EFFECTS
OF LEARNING SPREADSHEET MANAGEMENT
ON LEARNING DATABASE MANAGEMENT

Vágner Anikó, Zsakó László

Abstract: A lot of students learn spreadsheet management before database management. Their similarities can cause a lot of negative effects when learning database management. In this article, we consider these similarities and explain what can cause problems. First, we analyse the basic concepts such as table, database, row, cell, reference, etc. Then, we discuss differences from the point of view of the metadata. In our article, it is very important that we review differences which concern managing stored and calculated data. Finally, we mention sorting and filtering features.

Key words: computer science education, database systems, spreadsheet, teaching, relational data model

1. Introduction

Students in Hungary get acquainted with spreadsheet management earlier than with database management. They learn about a lot of features of spreadsheets in a simple way which will appear in the database management later. But in the case of databases these features are more complex, or maybe more understandable.

A question arises, what are negative effects of previous spreadsheet management knowledge on learning of database management. If they learnt database management earlier than spreadsheet, they would have another approach of features of database management systems (DBMS).

In this article we give a short summary what concepts or features can cause the possible negative effects, and why.

2. Literature review

You can find a few websites on which spreadsheets and databases are compared (Indika, 2011), (Kid Electric, 2011), but it is hard to find articles in which education or learning of spreadsheet and database management are compared. We can state that researchers who deal with teaching methods of computer science have not studied this topic. You can find articles which concern with education of database management and are written in the recent past, such as (Zheng, 2011), (Zhong, 2010). However, Kleiner (2014) states that there is no news in database education in the recent past. Moreover you can also find articles about education of spreadsheet management, such as (Yeh, 2011), (Frydenberg, 2013). But you cannot find articles in which the influence of each other is studied.

3. Concepts and features of spreadsheets cause negative effects on learning database management

Table and tabular

The first concepts are the table and the tabular. You can see that even the names of them are very similar (also in Hungarian). By the introduction of relational data model you cannot found the concept of table on tabular or table concept of spreadsheets. The tabular concept of spreadsheets is not the same as the table or relation of relational databases. Both of them are "tabular", and the students can think that the table in the relational model is similar to the tabular of a spreadsheet, but they are not the same. Of course, you can move a database table into a worksheet, and you get a tabular which has column names in the first row, and the data of the database table in the other rows.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Year	
2	Minsk	-4,5	-4,4	0	7,2	13,3	16,4	18,5	17,5	12,1	6,6	0,6	-3,4	6,7	
3	Bucharest	-2,4	-0,1	4,8	11,3	16,7	20,2	22	21,2	16,9	10,8	5,2	0,2	10,6	
4	Stockholm	-0,7	-0,6	3	8,6	15,7	20,7	21,9	20,4	15,1	9,9	4,5	1,1	10	
5	Budapest	1,2	4,5	10,2	16,3	21,4	24,4	26,5	26	22,1	16,1	8,1	3,1	15	
6	London	7,9	8,2	10,9	13,3	17,2	20,2	22,8	22,6	19,3	15,2	10,9	8,8	14,8	
7	Paris	6,9	8,2	11,8	14,7	19	21,8	24,4	24,6	20,8	15,8	10,4	7,8	15,5	
8	Barcelona	12,4	14,1	15,9	17,6	20,5	24,2	27,5	28	25,5	21,5	17	13,3	20	
9	Lisbon	14,8	16,2	18,8	19,8	22,1	25,7	27,9	28,3	26,5	22,5	18,2	15,3	21,5	
10	Rome	11,9	13	15,2	17,7	22,8	26,9	30,3	30,6	26,5	21,4	15,9	12,6	20,4	
11	Athens	12,9	13,6	16	20,3	25,3	29,8	32,6	32,3	28,9	23,1	18,6	14,7	22,3	
12	Valencia	16,1	17,2	18,7	20,2	22,8	26,2	29,1	29,6	27,6	23,6	19,5	16,8	22,3	
13	Malta	16,1	16	17,8	20	24,2	28,5	31,5	31,8	28,4	25,2	21	17,5	23,2	
14															
15															

Figure 1. *A tabular which is not correspond to a relational database table*

On the contrary you know that you can create other types of tabular in spreadsheets. Let's consider the tabular of Figure 1. You can store data similar way in a database table but this is not a relational table, in this way we do not suggest storing data in a database that way. Another problem with this tabular, that you do not know exactly what the meaning of numbers in the tabular is. In a worksheet you can give additional information to the tabular in other cells; however, you cannot do this in a database. In a relational table you would store these data with 3 columns: city, month and average temperature; Figure 2 shows the adequate form. In this way you will know what is the meaning of the number based on column names, in other words you have given additional metadata to the data. But do you know what unit of measurement the numbers have? Both cases you do not know, which implies that you need additional documentation of the table.

Let's consider the last column of the tabular of Figure 1. If data of it was calculated from the other columns, you would not store in the database, because the DBMS could calculate them every time when you need it. If data of it was a new additional information you would store it, but in another table, because logically this is another entity, and it is not suited to the table of Figure 2.

Moreover, the relational model in third normal form does not like to store names of towns and months so many times. Instead, it stores them in other tables, one for town names and another for months.

Based on basic concepts and our example we find that the first negative effects of knowledge of spreadsheet management on education of databases are:
- The meaning of table and tabular (if the students think that they are the same, it can cause a lot of problem).
- By designing a relational table you have to follow a lot of rules, whereas spreadsheets do not need them. The relational model follows the entity-attribute-relationship conception. A relation stores

attributes of an entity in a defined format. Spreadsheets can store more than one entity in a sheet or in a tabular; moreover the format is not exactly defined.
- Spreadsheets show immediately the calculated data, but you have to write a program code (we mean SQL select statement or use some other feature of DBMSs) in case of a database table.
- Third normal form relational model does not like redundant data, whereas in spreadsheet management there is no rule for it, and it does not come to you to store redundant data in another sheet.
- Metadata has to be stored other way in each cases.

	A	B	C	D
1	Town	Month	Average temperature	
2	Minsk	Jan	-4,5	
3	Bucharest	Jan	-2,4	
4	Stockholm	Jan.	-0,7	
5	Budapest	Jan.	1,2	
6	London	Jan.	7,9	
7	Paris	Jan.	6,9	
8	Barcelona	Jan.	12,4	
9	Lisbon	Jan.	14,8	
10	Rome	Jan.	11,9	
11	Athens	Jan.	12,9	
12	Valencia	Jan.	16,1	
13	Malta	Jan.	16,1	
14	Minsk	Feb	-4,4	
15	Bucharest	Feb	-0,1	
16	Stockholm	Feb	-0,6	
17	Budapest	Feb	4,5	
18	London	Feb	8,2	
19	Paris	Feb	8,2	
20	Barcelona	Feb	14,1	
21	Lisbon	Feb	16,2	
22	Rome	Feb	13	
23	Athens	Feb	13,6	
24	Valencia	Feb	17,2	
25	Malta	Feb	16	
26				

Figure 2. *Tabular which is correspond to a relational database table*

Mixed data types

Let's see the next example on Figure 3. Spreadsheets make it possible to store other data type in cells belonging to a column. On contrary, domain (or data type) of data belonging to a column in a database table have to be the same, you cannot change data type of only one field.

Figure 3 shows an example of tabular in which the second column stores data with different types, namely texts among numbers. In the example you expect a number of pieces in each field, which you want to summarize later, however in a sheet you can give such information why you do not know these numbers, such as "he is thinking about it", "unknown email", or "he has not answered yet".

If you choose number type for a column in a relational table the DBMS does not allow you to write there texts. Apart from numbers you can write there null value.

In a word, the next negative effect is to teach students not to store mixed data type in a column of a relational table. It can be a habit for them; it is hard to give it up.

To continue this question, the spreadsheets can calculate (summarize, calculate average, etc.) data with these mixed type that way that it does not consider the text fields. DBMSs have problem with mixed type data. If you choose text type for the column, the calculation will be difficult, in this way we do not suggest it. If you choose number type for this column, you cannot store texts in these fields. A solution can be, if you use two columns, one for the numbers where you can also store null values, and another one for the texts which give information about the reason why the previous value is null. In this way you can calculate on the number typed column (null values will not cause any problem), and get information from the other column.

	A	B	C
1	Email	Pieces	
2	xy@freemail.hu	1	
3	kissAntal@gmail.com	unknown email	
4	lakat@frg.lt	he has not answered yet	
5	gbela@thl.hu	10	
6	orszag@juh.com	he is thinking about it	
7	kbela@zrt.hu	20	
8	patfay@reka.lt	50	
9	**Total:**	**81**	
10			
11			

Figure 3. *Tabular which store mixed data type in a column*

A student who is perfect in spreadsheet management would never think about this solution, but a student who has not learnt about spreadsheets can learn faster this solution.

Metadata

In databases tables have names, column names, which you can consider as metadata. You cannot mix them and data in databases.

How can you give a name for a tabular in spreadsheets? Maybe you give a name for a sheet or you write a title to a sheet. The column names will appear in the first row of the tabular, and you can also give names for rows that way. This means that you confuse data and metadata. Moreover, the user is not forced to give names to them.

Data, calculated data and visualisation

In database management concepts you differentiate data, business logic and visualisation.

In databases you store the data. You can calculate data that way that you write queries, but in the most time databases do not store the results. Visualization of data is solved by another tool or application.

In spreadsheets you confuse them, on a sheet there can be data, calculated data (with complex expressions) and diagrams together. If you look at a cell, you cannot decide whether it is a stored or a calculated value. If a cell stores an expression, you see there only the result (if the cell is not the active cell). If the cell is active, and it stores an expression, you find the exact computational formula in the formula bar.

In databases calculated data are not stored, so you have to calculate them for example in a query.

Students cannot distinguish stored and calculated data when they learn spreadsheets, but they have to do it when they learn databases.

Data type and format

It is a similar topic as the previous one. The data and their visualisation is different in databases and spreadsheets. In spreadsheets you give data type and format together. This means that storing and visualisation of data happen together. In databases when you create a table, you have to give data type to columns. You can deal with the format when you show results to users.

Place of columns and rows

Logical level you can transpose places of columns in databases. Similarly, you can transpose places of rows. Of course on physical level DBMSs use a sequence of them, but they do not show it to users.

In contrary, if you transpose columns or rows in spreadsheets, what will happen with cell references? Maybe it will not refer to the original cells any more. You have to check them and may correct them. If you use diagrams or other visualisation features, they will also changes. Additionally, the background data file will also be modified by application.

Moreover, in spreadsheets you can reverse rows with columns, and the meaning of the tabular will be the same. You cannot do it in databases.

Referenced data

Spreadsheets use cell references in expressions. If you reorganize a tabular because for example you want to have a better look, and you move a referenced cell, the reference can miss the cell which you want. Cell references cannot use metadata, because in the sheet there is no metadata. A cell reference uses a place (depends on the storing), and if data move, the reference will not go with it.

In a database you refer to table names and column names and if you need only one row, you use a value of primary key. You do not need to know where a DBMS stores data; you have to know the metadata which the query or reference need.

In spreadsheets you cannot use primary key, because it has not similar concept.

Granulation

If you want to add or delete data to/from a database you have to consider rows; a row is the unit of a table. In contrary, if you want to do it in a spreadsheet, you have to consider cells; a cell is the unit. Of course both applications can modify only a cell or a field.

Serial numbers

In spreadsheets you like to use serial numbers. In databases there is no need to use them, you will not store serial numbers, and there is no mean. Instead, you will have primary key values, which are maybe not numbers. If you need serial numbers for a table, your visualisation application will add to results of queries.

Concept of database

Spreadsheets know the concept of database, but the meaning is not the same as database of the relational database model. The main difference is that the database of spreadsheets consists of only one table, but database of the relational model consists of a lot of table, which reference to each other.

Database functions of spreadsheets

These functions cause a lot of problems in learning of databases. Spreadsheets offer features for filtering and sorting a "database". Results of both features can stay in the original place, which causes that the storing and visualisation is confused again. If you locate results to another place, and later original data change, the filtered or sorted data will not change dynamically together with the original data.

If you filter data and results stay in the original place, calculated cells will consider not only the filtered data, but also the original data which are not visible.

By databases you write queries if you need filter or sort, and their results are visualised by another application. In this way these problems do not come up.

4. Conclusion

We have found a lot of differences between thinking and working with spreadsheets and databases. To summarize them:

- spreadsheets do not make difference between visualisation, calculation and storing data;
- relational data model has a lot of regulation, whereas spreadsheets do not know and consider them;
- table (tabular) and database concepts in the two approaches are not the same, but using and meaning are similar;
- metadata is not used and marked in spreadsheets;
- DBMSs consider a row as a unit, whereas spreadsheets consider a cell as a unit;
- data is referenced based on place by spreadsheets, whereas based on table name, column name and primary key (metadata) by database.

We do not want to say that learning of spreadsheets or databases is unnecessary. Both of them have a lot of advantage. However, based on this article we can state that deep knowledge of spreadsheets can cause drawbacks to learning databases. Based on these drawbacks we advise to separate the types of tasks of spreadsheets from the types of tasks of databases, which means that students would not be confused if they did not use a spreadsheet as a database management system.

References

[1] Indika (2011). *Difference Between Database and Spreadsheet*. http://www.differencebetween.com/difference-between-database-and-vs-spreadsheet/

[2] Frydenberg, M. (2013). Flipping Excel. *Information Systems Education Journal,* 11(1) pp 63-73. http://isedj.org/2013-11/

[3] Kleiner, C. (2014). No news in database education by all accounts: seriously? *Proceeding SIGCSE '14 Proceedings of the 45th ACM technical symposium on Computer science education*, 728-729. http://dl.acm.org/citation.cfm?id=2544237

[4] Kid Electric (2011). Using Spreadsheets and Databases in the Classroom, The Silicon Tower: Academia and Technology. http://silicontower.blogspot.hu/2010/02/using-spreadsheets-and-databases-in.html

[5] Yeh K.-Ch., Xie, Y., Ke, F., (2011). Teaching computational thinking to non-computing majors using spreadsheet functions. *Frontiers in Education Conference (FIE)*, F3J-1-F3J-5. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6142980&isnumber=6142695

[6] Zheng, Y., Dong, J.,(2011). Teaching reform and practice of database principles. *6th International Conference on Computer Science & Education (ICCSE),* 1460-1462. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6028905&isnumber=6028569

[7] Zhong, H. (2010). Innovating Practice of the Database Teaching. *Journal of Shenyang Jianzhu University Social Science*, 2010 (3), 379-381. http://caod.oriprobe.com/articles/24304793/Innovating_Practice_of_the_Database_Teaching_.htm

Authors

Anikó Vágner, Faculty of Informatics, University of Debrecen, Debrecen, Hungary, e-mail: vagner.aniko@inf.unideb.hu

László Zsakó, Eötvös Loránd University, Budapest, Hungary, e-mail: zsako@caesar.elte.hu

Acknowledgement