WCES-2010

The effect of visual query languages on the improvement of information retrieval skills

Jozef Hvorecký\textsuperscript{a} *, Martin Drlík\textsuperscript{b}, Michal Munk\textsuperscript{b}

\textsuperscript{a}Vysoká škola manažmentu/City University of Seattle, Panónska cesta 17, Bratislava 85104, Slovakia
\textsuperscript{b}Constantine the Philosopher University in Nitra, Tr. A. Hlinku 1, Nitra 94974, Slovakia

Received October 7, 2009; revised December 16, 2009; accepted January 5, 2010

Abstract

Using an experiment, two forms of query creation (SQL and QBE) have been compared from three points of view: the time period necessary for forming queries, their correctness, and their authors’ comfort during the formation process. The results were analyzed using advanced statistical methods. Some of them confirm popular expectations, while others indicate that the relationship between the approaches is more complex than expected.

In the conclusion, the authors indicate how their experimental results can be exploited for the users’ training improvement. © 2010 Elsevier Ltd. All rights reserved.

Keywords: Visual database query language; database end-user’s satisfaction; tacit knowledge.

1. Introduction

Programming-like activities allow people to transfer their ideas into commands executable by computers. In the terms of Nonaka-Takeuchi’s model (Nonaka &Takeuchi, 1995), programming serves as an example of the transfer of tacit knowledge into explicit knowledge. Such transfer is only possible if there is an appropriate notation understood by both sides – its sender and its receiver(s). The history of humankind shows plenty of appropriate notations for expressing tacit knowledge: alphabets for texts, music notations for melodies, mathematical expressions for calculations etc. They evolve with the progression of our knowledge in order to find an optimal way for codifying it. For example, the popular notation of integrals (\int) is about three hundred years old (Burton, 1988).

The history repeats with programming languages. Again, we are looking for notations expressing our intentions in optimal ways. “Optimal” may have different interpretations: requiring the shortest time, minimizing the number of potential errors, allowing the highest user’s comfort, etc. Due to that, graphic user interfaces (GUI) quickly replaced more traditional text-oriented communication. In fact, it has not disappeared; it is only exploited by a narrow community of IT professionals now.

Nonaka-Takeuchi’s model also stresses the fact that a part of humans’ tacit knowledge is codifiable; another one is uncodifiable (Nonaka &Takeuchi, 1995).

* Jozef Hvorecký. Tel.: +421 911 228 569; fax: +421 263 810 611
E-mail address: jhvorecky@vsm.sk

1877-0428 © 2010 Published by Elsevier Ltd.
doi:10.1016/j.sbspro.2010.03.090
For example, the fact that one is literate does not mean that he/she is capable of enjoying poetry and gets the same feeling as its author. The get it, one has to “read between the lines”.

Tacit knowledge is often divided into “hard” and “soft” one. The hard part can be transferred between individuals in its explicit form e.g. by books, manual or instructions. The soft part required understanding the context and can hardly be transferred in any other way than by human-to-human interaction. Again the reading a poem can be used as an example. If it is about skiing during a cold winter morning, it can hardly be appreciated by someone who never experienced a winter.

It implies that while coding, we put down not only letters but also ideas. A random sequence of letters or words hardly becomes a poem; random sequence of commands unlikely turns into a program. To form masterpieces, one has to be capable not only to memorize the commands but also to combine their potential into meaningful concepts. If commands express their potential in a more transparent manner, they invite more users. It explains the quick and overwhelming victory of GUI.

GUI belongs among so-called visual languages. A visual language is a set of practices by which images can be used to communicate concepts (Herkimer & Burnett, 1997). A visual language manipulates visual information or supports visual interaction, or allows programming with visual expressions. The latter is taken to be the definition of a visual programming language. (Najork & Golin, 1990). We'll try to avoid using the word "programming" when we don't mean to exclude non-programming visual languages. Educators can make two conclusions based on our above considerations:

- Teaching commands is not an equivalent to teaching programming.
- The learners will appreciate such language that will express its power in a transparent manner.

In this paper we apply the above concepts into teaching database retrieval skills. Traditional methodology focuses on information database internal structure and the syntax of SQL. Comparing with abovementioned knowledge management concepts shows that more attention is paid to the hard skills (i.e. to the operational role of SQL in database design, development and retrieval) than to the soft skills (i.e. to the formulation of users’ needs, comprehension of the function of data in reflections of reality, their exploitation and interpretation). As the hard and soft skills are inseparable, educators should prefer tools that combine them to maximum (Hvorecký & Drlík, 2009).

In our research described below we concentrated on information retrieval, in particular on the transfer of natural-language-formed user’s questions to queries executable by a database management system (DBMS). There are two basic tools for query formation: SQL and QBE (Chang, Costabile & Levialdi, 1993). Whilst SQL is an official standard and must be a part of every DBMS, QBE is rather a family of tools (sometimes called QBE-like languages). Its syntax is not exactly defined. It is based on a GUI and then translated to SQL – a feature that makes QBE “superfluous”. On the other hand, simultaneous existence of QBE and SQL indicates that developers of DBMS’s presume that QBE-like language is more user-friendly (Hvorecký & Drlík, 2007). Based on the authors' previous theoretical and experimental results, we formed a new methodology that includes QBE. The methodology was tested in university introductory database courses as we would like to develop the abilities in our trainees:

- To formulate their needs in an exact manner;
- To understand the necessity to transform them into a prescribed form;
- To decompose complex tasks into a series of simpler ones;
- To comprehend that the communication language (SQL or QBE) does not change with the application area.

It is impossible to compare the power of two complex concepts without discussing their separate features, their advantages and drawbacks (Sawyer & Mariani, 1995; Schneiderman, 1978; Vassiliou & Jarke, 1984). As we were interested in query creation, we split the problems into five groups G1 to G5 with growing complexity:

1. Simple queries in a one-table database requiring selection of data based on a property and/or their sorting (G1);
2. Selection of data from several tables a multi-table database (G2);
3. Aggregate queries in a one-table database (G3);
4. Aggregation and grouping of data from several tables in a multi-table database (G4);
5. Nested queries and subqueries (G5).
2. Experiment

The aim of our experiment was to compare the relationship between the programming tool and the behavior of its user. Thus, the query languages (SQL and QBE-like language) were the independent variables.

We were interested in the objective results as well as subjective feelings of their users during the query creation. For the factors (time, accuracy, and users’ satisfaction) became the dependent variables and were monitored (Bell & Rowe, 1992). They were measured in the following ways:

- Time needed for creating the query was measured separately for each task and student.
- Accuracy was expressed by a semantic difference between the correct answer and the one submitted by the student (Reisner, 1981).
- Satisfaction was quantified using 9-level Likert scale. The students express their satisfaction after completion all of their solutions.

The methodology was tested in a standard way using a test group and a control group. Each group contained more than 50 students with comparable knowledge and experience in information sciences. Our experimental plan included pre-test and post-test. The pretest confirmed equal levels of prerequisite knowledge in both groups ($p = 0.330751$). The results of the experiment were analyzed using linear models.

2.1. Verification of the Hypotheses

We have formulated tree hypotheses. The first two hypotheses were tested using a posttest consisting of the query creation and identical for the both groups. The third one was verified using the questionnaire. The hypotheses were analyzed using the one-way analysis of variance.

Our first hypothesis stated that the students applying QBE-based approach will need shorter time for query formation. The hypothesis has been proved ($F(1, 114) = 13.398, p = 0.000383$).

The second hypothesis expected that the QBE students will create more accurate queries than those using SQL. This hypothesis has not been confirmed. The correctness of students’ queries was almost identical in both groups ($F(1, 109) = 0.424, p = 0.516098$).

The third hypothesis presumed that the QBE students will feel better comfort during their query creation and higher satisfaction with their results. This hypothesis has been confirmed ($F(1, 114) = 12.779, p = 0.000515$).

2.2. The Impact of Time Periods on Query Correctness and Overall Satisfaction

The verification of the first hypothesis has proved that QBE queries have been made faster. To achieve a more detailed picture, we decided to include and investigate an additional factor – the time period necessary for the query completion. The analysis of covariance showed another statistically significant difference, different of our expectations. If the time for the solutions would be identical, the SQL group responses would be more accurate.

Based on ANCOVA results, the zero hypothesis – the presumption that there is no statistically significant difference between the groups in the precision of the query formation – was rejected at the 99% confidence level ($F(1, 108) = 11.125, p = 0.001168$). It shows that the relation between the covariate – the variable $sumT$ (total time) – and dependent variable $sumB$ (total score) – is statistically significant at the 0.01 level ($F(1, 108) = 70.169, p = 0.000000$).

Similarly, the covariance analysis demonstrated a statistically significant difference between the groups in the satisfaction of the query formation even if the time is taken into account. The zero hypothesis stating that there is no statistically significant difference between the groups when the time is taken into account could be rejected at the 95% confidence level ($F(1, 113) = 4.947, p = 0.028121$). It also shows that the relation between covariate $sumT$ a dependent variable $vpS$ (overall satisfaction) is statistically significant at the 0.01 level ($F(1, 113) = 19.046, p = 0.000028$). In this case, the analysis of covariance ($p < 0.05$) confirms the results of the analysis of variance ($p < 0.01$). Figure 1 illustrates the means with error plot showing (a) the correctness of the solution with respect to the mean time required for its creation and (b) the overall satisfaction of the user with respect to the mean time required for its creation.
2.3. Similarities and Differences among the Groups of Queries

2.3.1. Repeated Measures Comparison of the Groups of Queries

The most interesting results have been generated by ANOVA repeated measures. They helped us to show statistically significant differences between combinations of variables (time, accuracy, and satisfaction) and the level of task difficulty. Here, the division of problems into the categories $G_1$-$G_5$ played its role, too.

Based on ANOVA repeated measures three zero hypotheses were rejected at the 99% confidence level. They stated that (a) there is not statistically significant difference between the groups of query types in the time duration ($F(4, 456) = 81.218, p = 0.000000$), (b) in the precision of solutions ($F(4, 456) = 17.995, p = 0.000000$) and (c) in the students’ overall satisfaction ($F(4, 456) = 83.053, p = 0.000000$). In other words, there are statistical differences between particular types of queries.

In order to identify homogeneous groups of query types, we used multiple comparisons (Fisher LSD test) for their clustering in accordance to average time, average precision of solution and average overall satisfaction with particular types of queries (Table 1).

The analysis showed that the groups $G_2$ and $G_3$ are perceived identically in the terms of all three variables: time duration, precision of solution, and satisfaction. The next homogenous group is composed of $G_1$ and $G_3$; in this case, their time duration and overall satisfaction correspond. On the other hand, a statistically significant difference in the term of end-user’s overall satisfaction ($p < 0.05$) has been proven. The groups $G_4$ and $G_5$ create last identified homogeneous group of query types. The end-users perceived this pair of query types being same from the point of view of their overall precision of solutions but there was a statistically significant difference ($p < 0.05$) in the terms of time duration and overall satisfaction.

| a) | b) | c) |
|----|----|----|
| **Time** | **Score** | **Satisfaction** |
| Mean | 1 | 2 | 3 | 4 | Mean | 1 | 2 | 3 | 4 | Mean | 1 | 2 | 3 | 4 |
| G1 | 153.27 | **** | | | | | | | | | | | | | |
| G3 | 171.34 | **** | **** | | | | | | | | | | | | | |
| G2 | 188.96 | **** | | | | | | | | | | | | | |
| G4 | 265.11 | **** | | | | | | | | | | | | | |
| G5 | 306.76 | **** | | | | | | | | | | | | | |

Table 1. Homogeneous groups of the query types for a) time duration, b) precision of solution and c) overall satisfaction.

---

Figure 1. The means with error plot showing a) the correctness of the solution with respect to the mean time required for its creation; b) the overall satisfaction of the user with respect to the mean time required for its creation.
2.3.2. Comparison of the Groups of Queries Considering the Used User Interface

We have obtained other interesting results using Tukey HSD test (Unequal N). They refer to the relationship between the categories of tasks separately in the test and control groups.

The natural presumption that the more difficult tasks require more time to solve has been confirmed ($F(4, 456) = 19.323, p = 0.000000$) within the both groups. Figure 2a indicates the statistically significant differences between periods necessary for the query creation and selected combinations of time and category of the task. Each line corresponds to one of the groups.

We were also interested in the differences between time periods spent on the same tasks. The most significant differences appear between the groups $G_2$ ($p = 0.005693$) and $G_3$ ($p = 0.611760$) but a statistically significant difference ($p < 0.01$) has been proved in the case of $G_2$ only. It implies that the time for creating QBE queries with a single condition in multi-table databases is significantly shorter than that of similar queries in SQL. There is also an observable difference between the $G_3$ tasks (aggregated queries) but it was not statistically significant ($p > 0.05$).

Another statistically significant difference has been noticed between the precision of solutions and the query type in the both groups. The zero hypothesis stating that there is no significant difference was rejected ($F(4, 456) = 0.977, p = 0.019059$). Thus, there is a strong relationship between the precision of the solution and the task difficulty – see Figure 2b. The picture indicates a difference between some problem groups (QBE vs. SQL) – in particular $G_1$ ($p = 0.992792$) and $G_3$ ($p = 0.948587$) – but they are not statistically significant ($p > 0.05$).

Finally, we focused on the relationship between overall satisfaction of the users and the query type. Overall satisfaction of end-users is decreasing with increasing complexity of queries (Figure 3). One could anticipate this effect – its relevance to the subjective nature of this variable is obvious. If the end-user has to make greater effort to understand the merits of the task and used interface that does not offer sufficiently simple solution, his/her satisfaction would be on decline.

![Figure 2](image)

Based on ANOVA repeated measures results, there is a statistically significant difference between overall end-user’s satisfaction and different types of queries in the both groups ($F(4, 456) = 13.193, p = 0.000000$). There are apparent differences in overall satisfaction of the user in favor of QBE-like graphical user interface in tasks from the groups $G_2$ and $G_3$ (Figure 3). A statistically significant difference ($p < 0.05$) has been proved in the case $G_2$ ($p = 0.015005$).
3. Discussion

The experimental results described in the previous chapter uncover interesting relations between miscellaneous types of database queries from the end-user’s perspective of view. What are their consequences for planning our future training methodology? First, some results have been expected or “unofficially known” (Catarci & Santucci, 1995a; Catarci, & Santucci, 1995b). For example, we have proven apparent contribution of the QBE-like languages (or GUIs) for easier formulation of join and aggregation types of queries. On one side, the homogeneous groups G_2 (Selection of data from several tables a multi-table database) and G_3 (Aggregate queries in a one-table database) are perceived likewise in the term of time duration, precision of solution and overall satisfaction; on the other side, there are statistically significant differences between their graphical and textual formation.

The homogenous group G_4 (aggregation and grouping of data from several tables in a multi-table database) and G_5 (Nested queries and subqueries) represent a challenge to interface designers. Both require extensive efforts of their creators. They must first demonstrate substantial knowledge of logical thinking to specify and design an appropriate evaluation method. In the next step, they have to apply their algorithmic thinking to formulate final query in particular user interface. Thus, the main source of difficulty is not rooted in the choice of language (SQL or QBE) but in the semantic level of the task. The challenge consists of designing a graphic-oriented method that might simplify their creation.

Finally, we would like to comment the reliability of our outcomes. They could have been affected by several factors. First of all, the participants’ selection might be questioned. The group consisted of the students of our preliminary database courses. Their characteristics may differ from typical end-user in non-academic environments. On the other hand, other surveys in the same field (Chan, Wei & Siau, 1994; Chan, Siau, & Wei, 1997; Owei, Navathe & Rhee, 2002; Yen & Scamell, 1993) applied a similar approximation of “end-user” and their conclusions have been accepted as relevant.

The second factor is the number of the participants in the experimental and controlled group. Although it may seem sufficient, it is evident that with greater number of participants the results would become more general. The same argument applies to the repetition(s) of the experiment. We could not afford such numbers considering (a) the size of our institution (b) regular changes in the syllabus that exclude prolongation of the identical content over several consecutive years. The different complexity of the tasks is another factor that might have effect on the experimental results. From the very beginning, we understood that tasks leading to subqueries may affect the overall reliability of some results of our test. They are much more complex than other four types of tasks. Their extreme complexity could therefore affect the overall end-user’s satisfaction. As they are very frequent in day-to-day situations, we decided to include them to the test despite the above reservation.

We cannot forget in this discussion the limited duration of individual tasks as well as of the entire testing procedure.
Although we took sufficient free float for each task, the clock displayed on the screen could have affected participant’s activity and disturbed his/her overall satisfaction.

4. Conclusion

In this paper, two principal methods of query creation have been discussed. Based on the experimental results, one can conclude that QBE as a form of graphic user interface speeds up the process and leads to higher user’s satisfaction. The outcomes are of a similar level of correctness compared to those achieved by SQL. On the other hand, the differences are significantly different only for the medium-difficulty tasks. With an exaggeration, one can say that “the easy tasks are similarly easy, whilst the difficult tasks are similarly difficult”. This opens two challenges targeting the “user-friendly” end-user training:

1. If the trainees are not going to be frequent (semi-professional) users in the future, QBE is to be recommended as a more appropriate tool for their training. Our results indicate that they will likely progress faster and with higher comfort.

2. Designing and developing a graphic-oriented (QBE-like) tool for simplified creation of the most difficult tasks (in particular, of subqueries and complex aggregations) might be a welcome achievement that would help to both students and instructors.

References

Bell, J. E., & Rowe, L. A. (1992). *An Exploratory Study of Ad Hoc Query Languages to Databases*. Proceedings of the Eighth International Conference on Data Engineering.

Burton, D. M. (2005). *The History of Mathematics. An Introduction*. McGraw-Hill Science/Engineering/Math.

Catarci, T., & Santucci, G. (1995a). *Are visual query languages easier to use than traditional ones?: an experimental proof*. Proceedings of the HCI’95 conference on People and computers X.

Catarci, T., & Santucci, G. (1995b). *Diagrammatic vs textual query languages: a comparative experiment*. Proceedings of the third IFIP WG2.6 working conference on Visual database systems 3 (VDB-3).

Chan, H. C., Wei, K. K., & Siau, K. L. (1994). An empirical study on end-users’ update performance for different abstraction levels. *Int. J. Hum.-Comput. Stud.*, 41(3), 309-328.

Chan, H., Siau, K., & Wei, K.-K. (1997). The effect of data model, system and task characteristics on user query performance: an empirical study. *SIGMIS Database*, 29(1), 31-49.

Chang, S. K., Costabile, M. F., & Levialdi, S. (1993). Modeling users in an adaptive visual interface for database systems. *Visual Languages and Computing*, 4.

Herkimer, G., & Burnett, M. (1997). *Graphical Definitions: Making Spreadsheets Visual through Direct Manipulation and Gestures*. IEEE Symposium on Visual Languages.

Hvorecký, J., & Drlík, M. (2007). From questions to queries. *International journal of emerging technologies in learning*, 2(4), 13-19.

Hvorecký, J., & Drlík, M. (2009). *Enhancing Tacit Knowledge of Database End-users*. Proceedings of the ICL 2009 - Interactive Computer Aided Learning : The Challenges of Life Long Learning.

Najork, M. A., & Golin, E. J. (1990). *Enhancing Show-and-Tell with a polymorphic type system and higher-order functions*. IEEE Workshop on Visual Languages, Skokie. 215-220.

Nonaka, I., & Takeuchi, H. (1995). *The knowledge creating company: how Japanese companies create the dynamics of innovation*. New York. Oxford University Press. pp. 284.

Owei, V., Navathe, S. B., & Rhee, H.-S. (2002). An abbreviated concept-based query language and its exploratory evaluation. *J. Syst. Softw.*, 63(1), 45-67.

Reisner, P. (1981). Human Factors Studies of Database Query Languages: A Survey and Assessment. *ACM Comput. Surv.*, 13(1), 13-31.

Sawyer, P., & Mariani, J. A. (1995). Database systems: challenges and opportunities for graphical HCI. *Interacting with Computers*, 7(31), 273-303.

Shneiderman, B. (1978). Improving the human factors aspect of database interactions. *ACM Trans. Database Syst.*, 3(4), 417-439.

Vassiliou, Y., & Jarke, M. (1984). *Query languages: a taxonomy*. Proc. of the NYU symposium on user interfaces on Human factors and interactive computer systems.

Yen, M. Y. M., & Scamell, R. W. (1993). A human factors experimental comparison of SQL and QBE. *Software Engineering, IEEE Transactions on*, 19(4), 390-409.