Extended measurement of an information system's performance

KESZTHELYI András

Óbuda Univesity
Keszthelyi.Andras@kgk.uni-obuda.hu

Using Neptun as our scholar information system our everyday experience is that it has some annoying problems even today. Some of them are ergonomic, some of them are technical, others have theoretical roots. Perhaps the most annoying of these problems is that that it is usually too slow. This slowness can be very disappointing especially at the beginning of the examination term and at the registration period. In this paper I describe the circumstances of a measurement which shows that much better performance can be reached based on a good data model even in a poorer working environment and even in case of more serious demands.

Keywords: database efficiency, scholar information system

1 Introduction

The administration of the scholar records of the students has become a great task for today which needs great resources. This is caused not only by the increasing number of students but by the complexity of the credit system as well in which students can choose their own way of their studies.

To manage the administration by hand in such circumstances can hardly be imagined. Almost everywhere this task is solved by computer based information systems. Such a system can be called a scholar information system, or SIS, shortly.

In Hungary the market-leading system is the Neptun. There is another one, the so called ETR (Egységes Tanulmányi Rendszer, Uniform Scholar System). Our university has been using the Neptun for about ten years.

We have been experiencing problems since the beginning, problems of very different kinds. These problems - from the point of view of the end-users - can be grouped e.g. in the following way: problems of loadability, reality, ergonomics etc. How large number of administrative tasks can the system manage

simultaneously? Does the system work in accordance with the real life, with the administrative rules of the university? How many mouse-click is needed to perform the most frequent activities?

In this paper I examine the first mentioned problem, the loadability of a SIS-like database in a typically problematic situation, in case of the registration for examinations in an indirect way. I planned and executed an extended measurement to determine how many students and/or tasks can be performed almost simultaneously in harder circumstances.

2 Prior researches

Personal experiences are to be mentioned first, which are, of course, not part of a research but have played a great part in my motivation.

2.1 Student questionnaire

In 2008 I made a student questionnaire to investigate how our students rate the Neptun. I handed out 300 sheets containing 9 open questions and there were 273 valuable answers. Full details are in my PhD disstertation [1]. Some of the results are summarized below.

Table 1.
Fields that ought to be (very) different

nothing	21%
registration for courses and exams	21%
speed	13%
sending messages not only by login code	10%
timeout	8%
stability	8%

Table 2.
Usual and random errors

closes connection	44%
too little timeout	26%
nothing	15%
slow	10%
freezes	10%
data loss	10%
please wait	3%

Most of the mentioned problems are rooted in that the system cannot cope with higher loads so well as it could be expected. This circumstance can have many different reasons. I investigated the role of the three-level data modelling in an indirect way [2], [4]. In this paper I describe the circumstances of an extended measurement which shows that better performance can be reached based on a good data model even in a poorer working environment.

2.2   Role of data modelling

The role of data modelling, especially the role of the quality of the data model and its effect on loadability was investigated in a paper [2] at the 2nd International Conference for Theory and Practice in Education in the last year also by me. A data model can be considered good if it is a) understandable, b) unambiguous, c) realistic, d) full and e) minimal according to Halassy [3]. I pointed out that the adequate three level data modelling is a necessary but not enough prerequisite of the quality, included the loadability, of information systems. The requirement of minimality means two different things. First of all it means that the data model should be free of redundancy. Secondly, this means that all functions which are not needed to fulfil the original aims should not be implemented.

2.3   How to Measure an Information System's Efficiency?

In my last year paper [4] I described the circumstances of a measurement which was executed by a colleague of mine and me on a test database. This measurement shows that better performance can be reached based on a good data model even in a poorer technical environment.

It was an interesting problem to decide what to measure in order to get relevant results. Summarizing the results of the measurement it follows that the quality of the conceptual level data model ought to be considered the most important factor which influences the performance of an information system.

2.4   The role of the tools and methods

Péter Szikora described the test environment and application more precisely in his paper [5] last year. His conclusion was that (much) better results can be achieved if we choose the appropriate software tools not only for developing but for the operating environment as well.

2.5   The measured performance

The role and importance of the operating system, server softwares and other software tools needed to develop a client-server application are described in Measured Performance of an Information System [6] by P. Szikora. In this paper he describes the developing of the test application on which the first measurements were done and the proceeding of the load of the server during the measurement.

2.6   Teaching and learning informatics

Investigating the skills and knowledge of the students in the fields of programming and of database management we find an alarming situation. 98,7% of the students of eighth grade have never been taught to any programming language nor any algorithms, which is worse. [7] In the homeland of John von Neumann, in the informational age. Maybe it is natural that we cannot produce better systems (see Conclusions).

3   The extended measurement

3.1   Summary of the original measurement

I chose the registration for exams as a critical field to investigate from the point of view of server loadability. The two important questions we asked (and aswered) by the basic measurement were the following: Could the response times be tolerated by an average human student or not? How many of the registration attempts are fulfilled?

The borderline between the 'tolerable' and 'intolerable' time requisites cannot be defined exactly in a mathematical manner because it is a subjective opinion of the end-users, in this case the students. In the basic measurement the average and the maximum response times and the successfulness of the registration attempts were measured. If the response times and the number of the unsuccessful attempts are significantly lower than in case of the Neptun system, even in a poorer working environment, the importance of the quality of the data model and the difference of the data models are proven.

The test database contained 8192 students, about as many as our active students, four examinations for each of them. The number of places was one and a half times bigger than the amount of the students' examination. The registration itself

was made by PHP scripts. Each test user registered a date for all her (his) examinations successfully. Test registrations were started almost simultaneously.

The total time needed for the 32.768 registrations of the 8.192 students varied between 187 and 189 seconds (3 minutes and 7-9 seconds), so the average time was needed for a test student was 0.0228 seconds. The maximum value for the server load (1 min load) was 2.85, with an interesting, staircase-of-staircases like diagram as described in [6]. Repeating the original measurement in the same circumstances this year I got a result of 3 minutes and 6 seconds.

I was (and am) not in the position to make test measurements on the Neptun system, but I think that the above cited values are good enough to say that the importance of the quality of the data model and the significant difference between the two data models (the data model of the test database and that of the Neptun) could be stated as proven.

3.2   Need for an extended measurement

For the basic measurement the simplest possible data model was used. Only that part was implemented which was needed for the registration of examinations included, of course, the needed elements to check if a test student was allowed to register or not. Other parts of the data model was not developed and was not implemented in the test database.

In [4] I determined the most important factors which affect database efficiency, i.e. how a database can cope with high loads: hardware environment, software environment (included the operating system and the database management system) and last but not least the application itself. After choosing the programming language and tools there are two main fields which determine the performance of the developed program(s). These are the quality of the applied algorythms and the quality of program coding. In case of databases the 'algorythm' has a more special meaning than in general: the quality of the data model is included as the most important, necessary but not enough circumstance.

One might say that the measured results would be worse if the test database were a full (i.e. much bigger) one. So it is a reasonable question whether in case of a bigger test database the results would also be good enough to prove that the quality of the conceptual level data model is the most important factor in efficiency.

The size of a database, of course, can be denote in megabytes but it would be too simple. Naturally the more megabytes the size of a database the more time the same operation needs. More sophistically the main factors of size to be investigated are the following: number of tables, (average) number of columns in the tables, (average) number of rows of the tables, the number of indexes.

So it is reasonable to repeat the original measurement described precisely in [4] and [6] in case of a bigger test database.

4   The extended measurement

4.1   Increasing the size of the database

In this step the size of the test database was increased in order to have a reasonable size. The size of the Neptun database is said to be of more than 40 gigabytes at our university. The number of our students is about 12 thousand [8]. This means that each of our student is represented by a book of about 18 hundred pages. It seems to be absurd, the Neptun database must be very redundant.

The test database which holds the data needed to perform the registrations for examinations in a realistic way, i.e. checking all the prerequisites which are described in our statutes, has a size of 16 MB. Based upon this and taking into account the requirement of minimality (see section 2.2) the expected size of a full-functionality database might not be more than about 256 MB.

I extended the test database with 30 more tables. Each of them has 8 string type fields of 72 bytes (varchar(72)), of which the first is the primary key, a foreign key field, which refers to the students table with an index on it, and another index on the concatenation of three of the string fields. 8k rows was inserted into each of these tables.

The database size grew up to 272,8 MB. The registration for courses took only a bit more than the original time: 3 min 8 sec. The difference is less (1,08%) than the possible measurement error.

It seems that the total size of the database and the time needed to fulfil all the registrations are independent factors.

4.2   Increasing the number of students

I doubled the amount of students, so the test database contained 16.384 test students. New test students joined to the same courses as the original ones (i.e. the headcount of the existing courses were doubled also. Each test student (old and new) attends to 4 courses with examinations. The maximum number of students per examination were doubled, too. Database size: 285 MB (before registration). All the registrations were successful, exactly 65.536 rows were inserted into the appropriate table in 6 minutes and 16 seconds. It seems that there is nearly a linear correspondence between the number of students and the time demand. In an exact

linear case the time demand ought to be 6 minutes and 12 seconds. The 4 more seconds mean 1,07% which is also less than the possible measurement error.

5 The results

The results were only a bit better than I had expected before. I expected that in case of growing the size of the test database I would measure a significant growth of time demand, I would not have been surprised if I had measured about 150% of the original time demand. I supposed that in case of increasing the number of the test students the time demand would grow to 300-400% of the original one. Even so the repeated measurements say that the total size of the database and the number of the test students are not serious factors of the loadability. If 16.384 test students can register for their all 65.536 examinations in 6 minutes and 12 seconds it is a very good result. Better results could be reached if I were able to start more registration processes simultaneously (I measured 5 min 10 secs as well) so in this case the bottleneck is the client side.

To reach this result I did not use any special hardware or software or programming solutions. The database server is a PC which has an Intel processor of four cores and eight GB RAM. It runs Debian Linux, Apache 2.2.9 with PHP 5.2.6 and MySQL 5.1 as a Relational Database Management System (RDBMS). Choosing of the examination dates are also performed by the PHP script running on the server, not at client side. This means that the load of the server is a bit larger than it should be. Even transaction handling was not used, the whole table is locked for the registering user.

I must repeat at this point, that I did not use any special (logical or physical level) trick in order to increase the performance. There are such methods but I did not want to use them. I would have liked to show the importance of the conceptual level data modelling first of all. I think I have succeeded.

6 Conclusions

Of course I am not allowed to make measurements on the real Neptun system. So I cannot give numerical results of its loadability. Our everyday experiences and the student questionnaire shows that it is too slow especially in the most critical periods. Even so that our real students can start their real registrations by faculties on different days, so not more than about two thousand students ought to be served nearly simultaneously. In this case I think I can state that my results are much better than that of the real system even without measuring the latter exactly.

At this point I have two open questions. The first one: How is it possible that in the homeland of John von Neumann there was nobody in the last ten years who would have developed a better scholar information system? Maybe we haven't enough professors and students of good skills and knowledge in informatics to do that (see 2.6, Teaching and learning informatics). The second question: Why the first question has never been asked?

References

[1]   Keszthelyi András: Database based optimization in the higher education. PhD dissertation, Eötvös Loránd University, Budapest, 2010.

[2]   Keszthelyi András: The Role of Data Modelling in Information System Efficiency. In: Karlovitz János Tibor (szerk.): 2nd International Conference for Theory and Practice in Education – Teaching and Learning. Association of Educational Sciences, Budapest, 2009. p. 26.

[3]   Halassy Béla dr.: Az adatbázistervezés alapjai és titkai. IDG Magyarországi Lapkiadó Kft., 1995.

[4]   Keszthelyi András: How to Measure an Information System's Efficiency? In: 7th Conference on Management, Enterprise and Benchmarking, Budapesti Műszaki Főiskola, Budapest, 2009., pp. 213-219.

[5]   Szikora Péter: The Role of the Tools and Methods of Implementation in Information System Efficiency. 2nd International Conference for Theory and Practice in Education, Budapest, 2009.

[6]   Szikora Péter: Measured Performance of an Information System. 7th International Conference on Management, Enterprise and Benchmarking, Budapest, 2009.

[7]   Kiss Gábor: A magyar informatikaoktatás vizsgálata. In: AGTEDU 2008, ISSN: 1586-846x, (pp. 163-168.)

[8]   http://uni-obuda.hu/Bemutatkozik-az-%C3%93budai-Egyetem
download: 08-05-2010 (m-d-y).