

THE INFORMATION SYSTEM AS A SUPPORT OF THE EDUCATION PROCESS ON SCHOOL OF BUSINESS ADMINISTRATION OF SILESIAN UNIVERSITY IN KARVINA

VÁCLAV KRÓL

Silesian University in Opava

School of Business Administration in Karviná

Information Center and Systems

Czech Republic

Tel.: + 420 69 63 98 275

Fax: + 420 69 63 12 069

e-mail: krol@opf.slu.cz

Key Words:

database information system, Internet, user interface, client – server , network operating system, web server

1. Introduction

The information systems (IS) become an integral and certainly we can say one of the most important part of our life nowadays. The access to the informations plays the primary role determining the next decisions. So the IS results must be as fast and useful as possible. The paper tries to demonstrate the problems usually rising during the development and operation of the IS. It demonstrates it by describing the creation process of the IS on School of Business Administration of Silesian University in Karvina which almost all users who are involved in the education process at school in some way have to use daily.

In the half of 1997 the School information center was charged to construct a new database information system which was thought to replace the current system used from 1993. The new system was put into operation on October 1998 and today it covers all the decisive school requirements. It makes use of the user-friendly interface of the Windows95 operation system. Its great priority to the future is the connection of the data base with the Internet.

2. Starting conditions

The school has approximately 1800 students, 100 teachers and other 30 occasional users. The network operation system is Novell NetWare combined with free Linux. Client stations have Windows95 as the only one operation system. Because the rector's-office and the other faculty of the university are placed in another town, it was assumed the future possibility of mutual connection, but not necessarily on-line at all costs. Regarding to high expense on buying suitable and powerful database instruments from the very beginning the biggest stress was laid on an expense minimization. Of course the expense minimization problem is essential everywhere but in the educational system these problems are more specific in comparison with commercially ordered projects.

3. The architecture

3.1. Theory

The development of the information system architectures comes through series of models, cohering not only with the development of the technologies itself but as well in the context of the effort spent on the whole life cycle of the IS development and maintenance. As the priority we always find endeavours for the “software prefabrication” and its “reusability”, for the interaction security among different IS applications, for the extension and enabling changes. Today we can distinguish 4 basic models:

- one-tier monolithic architecture
- two-tier architecture client/server (possible modification using stored procedures)
- multi-tiered architectures (with data-centric or process-centric middleware)
- dynamic multi-tiered architecture client/server

Multi-tiered architectures allow dividing the software application functions into the following layers :

- Presentation logic as the user-interface
- Application (also called business) logic comprising the application functionality
- Data access with the transaction logic

While presentation logic is always on the client side, the data access layer and the application logic can be anywhere even the logic is in most cases divided into several layers.

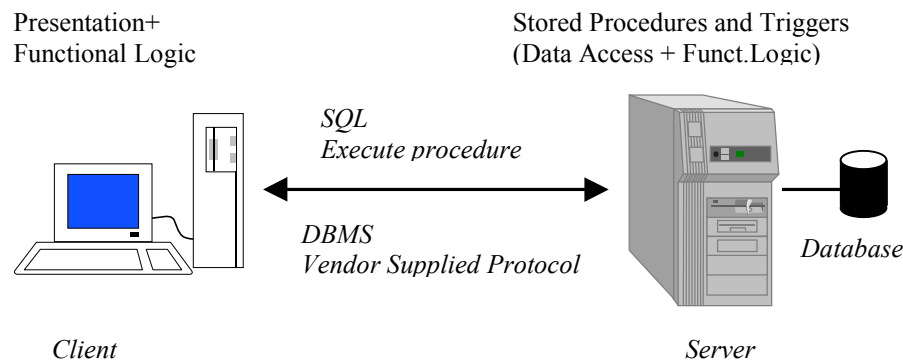


Figure 1: The model of 2 – tiered client/server architecture

3.2. Practice

For the new IS the classic two-tiered architecture client/server was chosen which was later advanced using the stored procedures. It is obvious that the architecture choice was done hand in hand with the choice of the database management system. Philosophically said the architecture is unidirectional with the pre-defined roles of the client and server. The client and the server are configured for mutual (point-to-point) communication. The client requires the database server services.

The growth of the applications built on the two-tiered client/server architecture is well possible if there is an enhancement of a shared data bulk on the servers. Against it, the growth of the demands on the application functionality leads to the performance advancement of the

client which becomes the “fat” client. Performance problems of this architecture also occur with the elevation of the transaction volume and during the live concurrent data access when the gained load of the transmitted data network between the server and clients is displayed. This was noticed at school during the “load hardest” operations such as the students’ schedule registration when there were more than 50 very fast working users. However, for the school needs setting the classic architecture is optimal.

4. Server side

4.1. Theory

There are many database management systems (DBMS) suited for the classic client/server database applications on the market today. Almost all of them are built on the relational model but recently the development is pointed at object-oriented model which is more natural. However it seems that the relational model will certainly be ahead from now on. From the developer’s viewpoint it would be preferable to have only objects everywhere and that is why sometimes the relationship “object-oriented access tool – relational database” leads to a paradox situation when the powerful relational DBMS are enhanced with a lot of stored procedures to a semi-object-oriented DBMS.

4.2. Practice

With a respect to the current network operation systems Novell and perhaps Linux it was naturally that we were looking for some free database management system which would be suitable enough for our requirements. The first choice was MySQL as the most popular DBMS in the whole free software world. But it has several disadvantages including the absence of the transactions and impossibility of the nested selects. So when SQL-server Interbase as one of the biggest DBMS appeared in the family of the free software products the choice was clear. Nowadays the SQL-server Interbase v. 4.0 is installed on a dedicated server with Linux operating system (OS). The SQL-server is under the GNU-license so it is free for our non commercial purposes. Interbase belongs to a family of the well known and used DBMS. It is supported on a lot of platforms (WindowsNT, UNIX, Linux, NetWARE). The created databases are transportable among these platforms. SQL-Server Interbase is noted for low maintenance and tuning requirements, on the opposite it is fully comparable to another great SQL-servers. By the help of the extended SQL language it enables defining of the stored procedures, triggers, user-defined functions, which means that some part of the application logic can be transferred to the SQL-server. Almost two year experience shows that the presented combination of the Linux as the robust operation system and Interbase as the SQL-server is a good solution for the IS server side even for larger range database applications. In addition Linux and Interbase are free.

5. Client side

5.1. Practice

The client application was developed in Delphi environment which belongs to a family of the modern instruments for rapid application development (RAD). The application was developed in the way that would enable a conversion to another data source without reimplementation with minimal costs which was the biggest stress from very beginning. This

involved a lot of work on the application background and SQL-query formation to satisfy it. From the current viewpoint it can be evaluated as a good idea of course but the later operation showed too useless overhead of the database engine as the connecting element. As the interfaces unifying is still in evolution it seems that the native data access of the classic client/server applications would be the best choice for its performance.

The another stress was laid on a user's comfort so the application today makes possible such things like individual adjustment of the program environment for every user in accordance with his needs. This means individual settings of the filters, printing reports, columns of the tables including the colors and font type, order settings and many others and all the settings can be saved. Surprisingly so sophisticated system with a great number of the miscellaneous options appears to be too complicated for the users and almost every user makes use of only a small percent of the possible settings. So the old good slogan "the strength is in the simplicity" appears as more suitable.

5.2. The relationship "User – Information system"

The relationship of the user and IS appears to be an interesting problem from the viewpoint of their habits and real needs in accordance with trends of modern information systems. The relationship is of course executed by the user-interface. Surprisingly current systems are mostly still developed in the classic form forcing the users during the work with the data records to keep the following order :

- action choice (for example students list)
- showing the dataset records
- searching the record I want to work with
- working with the searched record

It is naturally obvious that the point No. 2 can be fully omitted in most cases during the work with datas. This is not even possible but necessary from the viewpoint of the architecture client-server where the server is preferably intended for serving results of the requirements with the minimization of the network transmissions. I assume that in this point great speed reserves can be found in most of standard applications approaching remote data. Obviously the suitable worked user interface and well solved searching mechanism play the most important role. However, it does not need to be a problem thanks to the current RAD (rapid application development) tools, all depends only on programmer's invention and on the opposite on programmer's laziness because the suitable design of the cooperation between a man and a software product often requires much more work than the application logic itself. Then showing all records is useful only for displaying some summaries, summarizing reports and others. Although permanent escalation of the data network transmissivity assists in building applications where the users can browse through thousands of records (and surprisingly many of them require it), the complication and consternation occur along with the first attempt of the connection through the modem or another data narrow place. And by the way the most vivid example of the communication type "tell me what you want and I serve it" is popular browsing through the Internet by the help of the www browsers which always displays only the necessary information amount and waits for the next user input. This is also an example of the architecture client/server where the www browser has the client role sending the requirement (URL) to the server which serves the html document back.

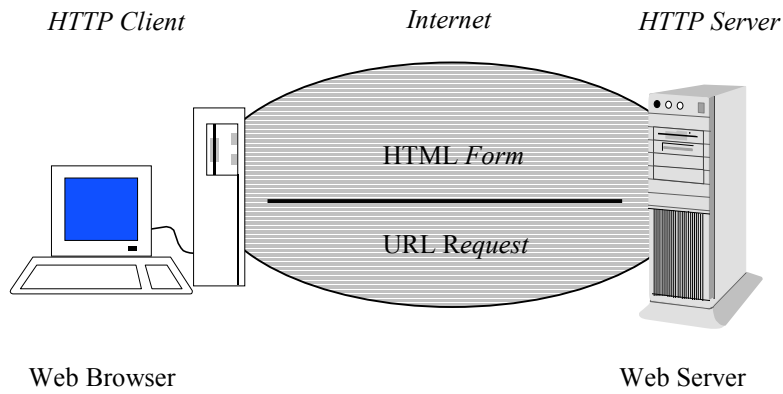


Figure 2: The simple “browsing for content” model of 2-tiered client server architecture

6. Future perspectives

6.1. Theory

Recently we can observe huge change of the type of the coming informations on the Internet. Not long ago most of the www informations was static – there were available only simple documents linked by hypertext links. However access of this type is obsolete nowadays. The web users require pages that offer an interactive access to the informations of any kind, for example to the informations from a database they work with during the day.

Of course this is connected with a necessity of system architectures’ changes and in this point the future belongs to the dynamic architectures. There will be sent not only the datas over the network but as well the applications and this will establish the dynamic model of the application tiers transfer over the net. Dynamics of the architecture is a feature that did not come from the IS creation environment but it grew up from the need of global communication integration over the Internet and the information systems. Dynamic features of the current architectures are perhaps expressed as an applets distribution on the network.

Many applications were developed along with the classic client/server architecture and they must function in the internet environment. So what are the possible solutions? We can choose among the following ways:

- *Using scripts*
Using scripts like CGI, ASP or PHP represents the standard data access solution. It is especially useful when we want realize simple queries without another software expenses. The PHP scripts using along with the Linux is the most used combination today (as PHP is free).
- *reimplementation of the current database application*
It is the more complex and permanent solution to develop the whole application in the www environment. Reimplementation is well possible if the current client/server application is implemented using the www-ready instruments. Reengineering is the right way if we assume further development.
- *Modification of the web server*
It means the web server functionality extension which the web server forces to be a communication mediator in the chain browser-web server-database server.

- *Modification of the web browsers*

It means the web browser functionality extension adding another protocol that enables the straight connection to the database server without using the web server. It is done by the help of plug-ins, Java applets, Java scripts or ActiveX objects.

6.2. Practice

Using PHP scripts along with the Interbase and the Linux OS seems to be a suitable solution nowadays. It assumes writing a lot of scripts and such functionality as in the classic application can not be obtained but the free PHP benefit in accordance with the used configuration is a good decision.

7. Conclusion

A great attention during the development of the client applications as the part of the client/server systems must be paid on a well designed user-interface. It is closely adherent with the result performance of the application and of course with the user satisfaction. The experience is the most important role behind it.

Although the Internet applications raising a great boom today the classic systems are still in progress and sometimes they are the only suitable for current requirements. But the trend to the Internet is unique. There are several ways how to migrate the current applications there. Usually it is done using the scripts but the reimplementaion is the more complex and permanent solution.

References

- [1] Hujňák P.: Směrem ke kooperativním informačním systémům. Sborník DATASEM'97, Brno, 1997.
- [2] H. Ladányi: SQL Unleashed. Sams Publishing, 1997.
- [3] M. Cantú: Delphi Developer's Handbook. Sybex, 1998.
- [4] J. Salemi: Databáze klient/server, Unis publishing, 1993.
- [5] M. Straka: Vývoj databázových aplikací, Grada, 1992.