

Database in Web Environment

Radek Dvorak, Ivan Jelinek

Abstract: The goal is to find a universal model of a data source for accessing heterogeneous data spaces and heterogeneous types of information on the Internet/web. The key area of data sources of today's world are databases. However, to access them is very difficult due to the variety of their nature and the inconsistent approach to data and their accessing. That is why a well created model of a data source will enable an easy and mainly consistent approach to the data sources both saved in the classic (relational) or specialized databases (object, XML, etc.), and other data sources. It is the interlink between the data source and the user (man, computer), which will remove the inconsistency of accessing data and will secure a user-friendly tool for cooperating with the data source. The formalism of the model will include also nonstandard methods of search, such as general search, contextual search, search for non-textual information.

Key words: Computer Systems and Technologies, Data Model, Database, Data source, XML language, Web application, Programming language.

Evolution of web technology

Under the term „engineering“ is generally understood the systematic application of the scientific knowledge during design and creation of the economically effective solutions of the practical problems. If we apply the engineering definition to the web area, too, then we can define the term web engineering as an application of the systematic, disciplined and qualified approach to the development, operation and maintenance of the web applications.

Web itself has its evolution, see the picture, from using the TCP/IP connection – static web presentation, through the interactive web presentation based on the client-server technology, to the presently dynamically developing area of web services. The last developmental stage takes into consideration not only the man-machine interaction, but also the machine-machine one. That is why great stress is put on the quality of the information description, so that both the man, and the machine can interpret it correctly. (Technology: WSDL, SOAP, UDDI)

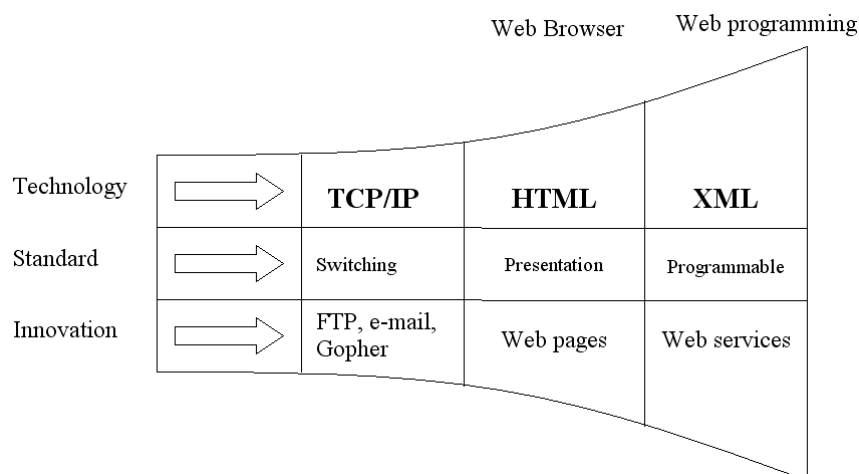


Fig.1 The phases of evolution of Web technology

TODAY'S WEB APPLICATIONS

The basis of the present web applications is a three-layer model. The web-client interface is considered as the first layer and the web viewer (Internet Explorer, Mozilla, ...) is most spread of them. The client cooperates with the second layer that is a web server

relevant to the web application. The web server supports the application programming (CGI+API DB, PHP, ASP, ...).

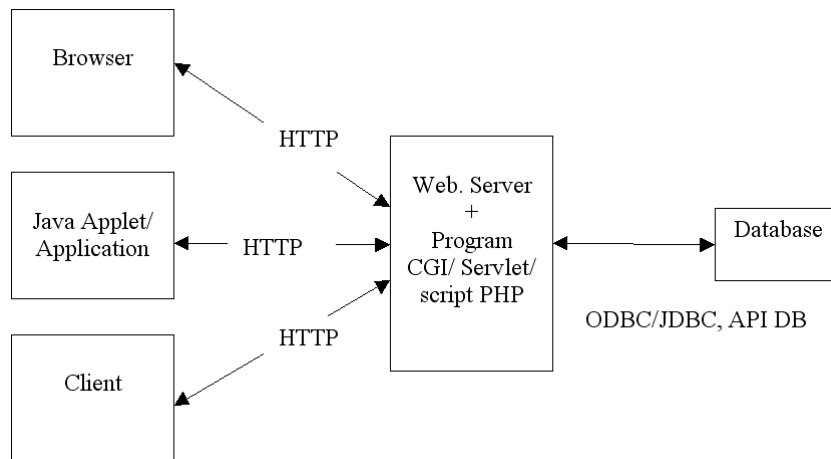


Fig. 2. The model of today's web applications

Application operating in the second layer needs to cooperate with some data source stored either in the form of „real“ database systems or non-structurally in various formats and on various platforms. Then, the third layer is mostly database system in which data information of the application is stored. The communication between program logic and a database system is presently provided by the ODBC, JDBC or by the special API interface of the relevant application interface to the database system.

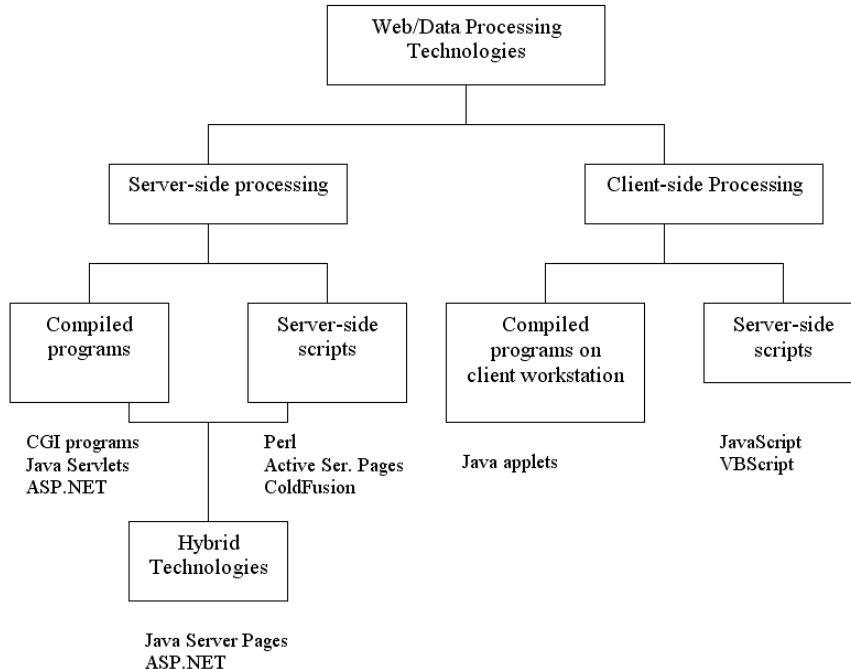


Fig.3 Division of programming technology in the web environment

The programming tools used for programming in the web environment are described in figure 3. The web applications can be divided to processes running on the side of the server and on the side of the client. Each of these areas is further divided to compiled programs and scripts. Today the hybrid technologies are being more and more enforced.

They combine the good characteristics of compiled programs and scripts. The most well-known examples of all languages are stated in the figure. These languages mostly enable the access to databases, either by using ODBC/JDBC, or by the specialized API for the relevant database.

DATA SOURCES

In today's world, where a huge quantity of information arises in every moment, the technologies are inseparable tool for the effective data storage, processing and re-searching. Based on the information technology and above all database evolution, several types of databases arise. Undoubtedly, the relational databases, termed as RDBMS (Relational Database Management Systems), are of the most spread ones. For example, MySQL (<http://www.mysql.com/>) can be indicated as their representative. The object databases, termed as ODBMS (Object Oriented DBMS), are the second (no less important) group. The GameStone database is their representative. But today, both groups begin to penetrate and present databases try to accept the better features if both groups they are termed as ORDBMS (Object Relational database). They can be represented by the Oracle database (<http://www.oracle.com>), that enables both relational and object data access with its advantages and imperfections. Beside the basic and most known types of databases, there are a great number of systems that solve data processing by their own way. As an example, we can name the Cache database (<http://www.intersystems.com/cache/technology/>) that is termed as post-relational database. Cache is a data storage that is able - under specific circumstances - to access the data by relational and object way. However, the XML databases with their native XML structure are developed presently.

Apart from databases, that are the hugest data storage, it is possible to store the data into the specific structure files, for example HTML, PDF, XML (the most number of formats are standardized by the W3C organization) etc.

In summary, we will call all the above stated ways of data storing data store. The data store group will never be final, because constantly new ones are being created and other cease existing.

Aside from the above stated differences between the individual data stores, it is still necessary today to take into consideration the platform above which the system is built, not only as an OS (operational system), but also a "protocol", through which the data are accessed.

And what is, at all, the reason for accessing information on the web through databases. The reason is very simple and is based on the basic specificity of web engineering. We want data to be safely stored, to be up-to-date, and their update to be done without the administrator's intervention. And it is the databases on the web that bring all these advantages.

When both the areas are connected, however, the problem of securing *the universal* access to data stores from the web, or the Internet environments seems critical. Today's trend for solving the situation of data accessing from the web is to use the ODBC and JDBC interfaces, however, there are significant problems connected to it not only in the limitation to only the basic commands of the SQL language (today the entry of commands almost on each database source differs), but also these solutions are not completely effective due to the necessity of accessing through other tools. In case of the JDBC interface, there are 4 ways and some of them require accessing through JDBC and ODBC simultaneously. Today for each data store there is in most cases already an API – a direct interface enabling the access to data, however, these interfaces are very specific and support only various levels of data access from the given programming environment to the relevant data source. The present research tries to achieve a local solution of the partial problems, such as for example the transfer of static pages to the databases, however, a

global approach is still missing. Today at specialized establishments new tools for bringing closer the various contents of the data stores are being developed; it is mainly searching for an effective transformation of one type of a data store to another while preserving the advantages of the relevant data sources. The above points suggest that there is no universal model, through which it would be possible to access data easily and without the knowledge of the data store specificities.

The goal of the research is to find a universal model of a data source for accessing heterogeneous data spaces and heterogeneous types of information on the Internet/web. The key area of data sources of today's world are databases. However, to access them is very difficult due to the variety of their nature and the inconsistent approach to data and their accessing. That is why a well created model of a data source will enable an easy and mainly consistent approach to the data sources both saved in the classic (relational) or specialized databases (object, XML, etc.), and other data sources. It is the interlink between the data source and the user (man, computer), which will remove the inconsistency of accessing data and will secure a user-friendly tool for cooperating with the data source.

The first step of the proposal was the carrying out of the analysis of the presently available database systems. In the first phase it was necessary to find database machines. The Internet offered quite a big amount of systems and projects, which solved these questions, however, at a closer look at these systems, one came upon a non-existent reference or zero information about the project, which was not finished. After filtering out these systems the database of "databases" was created. Its structure was chosen with regard to the analysis of the possibilities and requirements of today's systems. It is mainly the basic group of supported data types, supported interfaces, supported system environment and other additional information. The database of data sources is an open system, in which it is possible to update the information, and it is available at the following address <http://db.dvorakconsulting.net>. This analysis clearly shows the standardization of data types, the interface of smaller databases to the big ones that set the direction of the development.

THE RESEARCH RESULTS

The support of the programming environment for databases is in most cases very limited. The relational databases often support only PL/SQL. The object databases mostly support only their own native tools (e.g. for Cache it is the Caché ObjectScript, Caché Basic).

The interface to the database is a not a less important question, among the standard ones there is only the support for the ODBC and JDBC interfaces. Aside from these interfaces, the support for many databases is in the Java, .NET, C++ and PHP languages, mostly they are native tools. XML:DB API is the most widespread interface for the XML databases.

The SQL support in relational databases today is mostly only on the 92 version level. However, there are databases, which do not support even this one, some systems support selected extensions according to the 99 norm. The analysis of the supported operational systems shows the variety of today's IT world, in which databases are implemented on most of today available implementations.

Most of the network databases support the TCP/IP protocol, some of the systems also support the IPX/SPX Novell network environment.

As for the database type, the relational databases are still at the top with almost 60%, the XML are the next big group, followed by the combined (13%) and object databases (9%).

SETTING CONDITIONS FOR THE UNIVERSAL MODEL – DATA IN DATA SOURCES

The previous analysis brought the following results for the data elements. Data, which are contained in databases today, can be divided to two basic groups: dynamic and static. The basic elements of the dynamic ones are the methods and procedures – algorithm and “knowledge” – logical rules. Among the static data there are the “classic” ones, such as for example number, char, ... Today it is good to distinguish the types of the classic data, whether they are the atomic or the structured data types. The other significant group aside from the “classic” data is the texts, visual components (bitmap and vector graphics and video), and finally also the sound data types.

It is apparent from this enumeration that the variety of today’s data types is great and the access to them and work with them is not quite consistent. Each of these types requires the user’s knowledge of it and of working with it.

The perspective solution seems to be the creation of a model of data sources, into which it would be possible to save/record all such information, and then to access them without the knowledge of the relevant data source. The idea should be similar to that of the Java language, where the code is also translated to a “semi-finished product” byte code, which is then arbitrarily transferable to all platforms.

Today XML seems to be such structure; it both supports a great number of data elements used in databases, and also meets other conditions, such as an effective access to data for the human brain and also for the machine. The XML structure also enables a very sophisticated access to data, due to the XML tree structure. With the mass expansion of XML, the users become aware of the way of interrogating these data. Another reason for choosing XML is its ability to use additional information to various data elements, the supporting information of the model, which the XML structure also enables, and which is not quite easy to be recorded in the database systems.

This choice also very much reflects today’s trend of data storing, which is solved

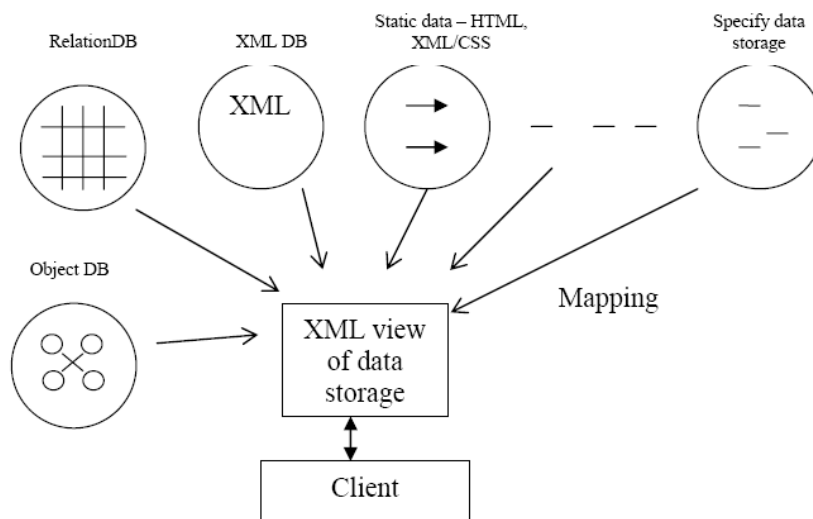


Fig. 4 System of a universal data model

either by storing itself in the database, or there are a lot of projects today, which seek the optimal methods of data transformation/storing to the XML documents, which, however, the model will accept. The analysis of today’s systems shows that their development converges to these ways.

The XML structure is also very suitable due to its great self-descriptive ability.

The possibility of an extending search would also depend on the choice of the structure and of the content of this tree. Not least, the XML language is well supported, when the information is displayed through a web interface.

CONCLUSIONS AND FUTURE WORK

The next goal will be to suitably propose the structure of the universal XML interface (XML picture of the data source), into which the present sources would be recorded, but the model should be general enough to enable the new types of data source to be incorporated in it.

To implement the recording and to test the unification of access of selected sources.

At the same time to take into consideration the local specificities – to consider not only the functional dependencies, but also the psychological perception of a human brain when accessing data.

REFERENCES

[1] CHAUDHRI, AKMAL B. Web, Web-services, and database systems : NODe 2002 Web- and database-related workshops Erfurt, Germany, October 7-10, 2002 : revised papers Berlin : Springer, 2003, pp. 206–220.

[2] KALINICENKO, L. A. Advances in databases and information systems : 7th East European conference, ADBIS 2003, Dresden, Germany, September 3-6, 2003 : proceedings, Berlin : Springer, 2003

[3] Murugesan, San, Deshpande Yogesh. Web Engineering, Springer-Verlag Berlin, 2001

[4] Carbenell J.G., Siekmann, J. Managing WEB-Based Data, IEEE IC, July August, 2002

ABOUT THE AUTHORS

Ing. Radek Dvorak, PhD student, Department of Computer, Faculty of Electrical Engineering, Czech Technical University, Karlovo namesti 13, 121 35 Prague 2, Czech Republic, Phone: +420 606 837 849, E-mail: Dvorakr1@fel.cvut.cz, <http://cs.felk.cvut.cz>, <http://webing.felk.cvut.cz>.

Associate Professor Doc. Ing. Ivan Jelínek, CSc., Department of Computer, Faculty of Electrical Engineering, Czech Technical University, Karlovo namesti 13, 121 35 Prague 2, Czech Republic, Phone: +420-22435-7214, E-mail: jelinek@fel.cvut.cz, <http://cs.felk.cvut.cz>, <http://webing.felk.cvut.cz>.