Proceedings of the 17th International Conference on Manufacturing Systems – ICMaS
ISSN 1842-3183

University POLITEHNICA of Bucharest, Machine and Manufacturing Systems Department
Bucharest, Romania

APPLICATION OF A WEB-DATABASE IN MANUFACTURING SYSTEMS

Aldeniz RASHIDOV

Abstract: *This report presents architectures of the connection of a Web application with database (DB) used in manufacturing systems. The offered extended architecture makes possible the data to be analyzed and processed, to take decisions about the running and control by the use of a previously created control algorithm. This algorithm uses the data and the opportunities that refer to the processing of the data from the DB and allows a great number of events that follow to be taken into consideration and solved directly by the database management system. There is a sure opportunity for device and control of the given commands to end-devices (ED) by the DB, used in an industrial environment. That makes the architecture safe and reliable.*

Key words: *web-database, industrial systems, control systems, web-based systems.*

1. INTRODUCTION

The development and the advantages, given by the global Internet, are a prerequisite for the projection and processing of Web based systems for collecting and processing data, received from different end-devices (ED), which work in an industrial environment [1]. During the recent years there has a particular interest in the systems, which have not only collecting and processing functions, but which realize a monitoring, analyzing and control functions on industrial processes through the Internet. These systems, known as Web-based information-running (control) systems, take some of the leading places among the services, which the Internet offers.

The main tasks, which they complete, are:

- Work in a real time;
- Monitoring, collecting, processing and analyzing the data, received from the ED via the Internet;
- Decision for control and control of the ED trough the Internet;
- Service of a simplified users` interface, protected from an illegal access;
- Choice of a sure and reliable architecture of the connection between a Web application and database (DB) via the Web.

The last listed task concerning a safe and reliable architecture of the connection between a Web application and DB through the Internet has a particular importance. The DB for Web is a diversion from the traditional ideas [2-4]. It allows to present the data by some objects (containing data and methods for manipulation of the data), as well as ensure access to them for each users, having the necessary access rights.

The information, generated by the DB for Web is independent from the program applications. DB for Web has a range of advantages:

- Receiving a actual information in time;
- Better opportunity for interaction with other users and organizations;
- Integration between the application, which use DB in Web and other applications;
- Access to remote industrial systems – monitoring and control through the Web.

The necessary resources for realizing access to DB trough the Web in the industrial systems are the following:

- Database for Web – Web DB;
- Program applications and methods, ensuring the DB access and necessary for the control remote stations;
- Program applications and methods, realizing a connection between the DB and the controlled devices;
- Web - server, serving the programs.

The functions connected with the collecting, processing, monitoring, taking the decisions for the control and the control are usually given by the programs which realize the access to the DB. Their effective and successful executing to some extent depends on the correct and exact choice of a method of access to the DB through the Web. The actual execution of the functions of collecting, processing, monitoring, taking running decisions and control is done with the help of program drivers. These drivers communicate at a lowest level with the ED. Dynamic and sometimes time-critical functions require the choice of an appropriate and reliable mechanism for asking the drivers.

The aim of this work is to present and offer architectures of the connection between program application and DB via the Web, used in manufacturing systems.

2. ARCHITECTURES OF THE CONNECTION OF A WEB APPLICATION WITH DATABASE, USED IN MANUFACTURING SYSTEMS

2.1. Architecture of the connection of a Web application with database

Here the Web-browser is a connecting point for accepting and sending of requests and results. The architecture of the connection of a Web application with DB in

the manufacturing systems consists of a user's Web – browser, a Web-server, a program interpreter, a driver for a communication with devices, DB and an ED. The data manipulation is realized by using scripts, which are started at the Web-server or at the user's browser. Fig.1 presents a block-scheme of architecture of the connection between a Web application and DB, in which the Web-browser is a connecting point for accepting and sending requests and results. This architecture contains two main branches, named {A} and {B}. The branch marked as {A} includes operations with the data accepted from the ED at circle {B} and concerning the storing, processing, selecting, analyzing and taking the decisions for control.

One transaction to the DB at the branch {A} consists of the following steps:

- A remote station sends a request {A} to do program module via a Web-browser;
- The Web-server finds the code of the program module and sends it {A2} to the program interpreter, to be processed and completed. The interpreter is a part of the Web-server, after it has been initialized as its module;
- The program interpreter analysis the program module. In a case that it contains operation for realization of a connection with the DB, as well as searching and processing some data in it, the program interpreter realizes a connection with the DB and sends separate requests {A3}, written in the language mechanism, served by the DB - SQL;
- The DB server accepts and stores SQL request. It processes them and then sends the results {A4} back to the program interpreter of the software support;
- The program interpreter finishes the execution of the program module, which usually includes synthesizing and formatting the result as a code, appropriate for interpretation by a Web-browser {A5};
- The Web-server sends that code {A6} to the browser and the user has a result.

The branch marked as {B} in the architecture is designed to do operation with an ED to collect data and control. A transaction to the DB here consists of the following steps:

- A remote station sends a request {B1} for executions to the program module via a Web-server. It is possible for the program module to be the same as the one that has been sent for execution along branch {A} and in it there is a function for requiring another outer program module (driver or etc.). In that case a request for execution of a combined program module (CM) is sent;
- The Web-server finds out the code {B2} of the program module - a driver for a connection with devices and executes it. In case when a CM is used, it is send {A2} to the program interpreter for processing and executing. The function for calling an outer module included in it finds out and executes a driver for a link to an ED {B2a}.
- The driver ensures access to a ED and sends commands {B3} for data collection and control;
- The EDs execute the received commands and send the {B4} result-data to be stored, processed and analyzed, or report the status of execution. Depending on the command, which has been given, the driver sends {B5} data for storing in the DB.
- During the control the driver is able to execute commands connected to the finding out and processing data {B5, B6}. In a case when a CM has been used, these commands are usually given by the program interpreter {B3, B4}.
- The status on execution of the commands is sent to the Web-server {B7} or the program interpreter {B7a}. When a CM is used, the program interpreter finishes the execution of the program module and formats the results under a code suitable for interpretation by the Web-browser {A5}.
- Depending on the initial request of commands, a generated by the driver code {B8} is sent to the browser and the user receives the result of the completed commands. When a CM is used, the Web-server sends the interpreted code {A6} to the browser and the user has the result.

2.2. Architecture of the connection of a Web-application with database, where the Web-browser prepares a request for the database

Fig. 2 presents a block-scheme of the architecture of the connection of a Web-application, where the Web-browser prepares a request for the DB. This architecture is different from the previous one only by the fact that in this architecture the user's Web-browser gives itself the programs which have to be executed to the program interpreter (after it has received them from the Web-server).

2.3. Extended architecture of the connection of a Web-application with database in manufacturing systems

The architectures, which were presented in the previous section, are most frequently used ones in the Web for realization of a connection between the program application and the DB [5].

These architectures offer access to the DB in order to realize the operations concerning, storing, processing, selecting, analyzing the data, as well as taking the decision for the control and control.

However, they have, some drawbacks:

- Taking decisions for the control and control are done directly through the user's interface. A processing or analysis result is sent to the user's interface and the user takes a decision about the execution of particular operations. There is a possibility for some critical in time operations not to be successfully executed;
- Each remote station can send commands via a Web-browser having the access rights. The arbitrary passing of commands, without distribution and control to ED makes conflict situations.

Fig. 3 presents an extended architecture of the connection between a Web application and DB, used in manufacturing systems, which shows how the mentioned drawbacks could be avoided.

It consist a user's Web-browser, a Web-server, a program interpreter, DB of MS SQL Server, a driver for a connection to devices and ED.

The architecture consists of three independent branches, named {A}, {B} and {C}.

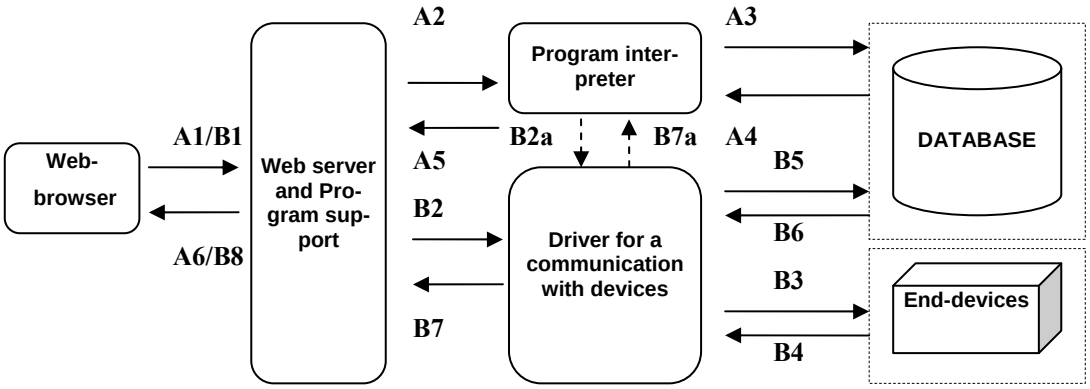

Fig. 1. Block-scheme of architecture of the connection between a Web application and DB, in which the Web-browser is a connecting point for accepting and sending of requests and results.

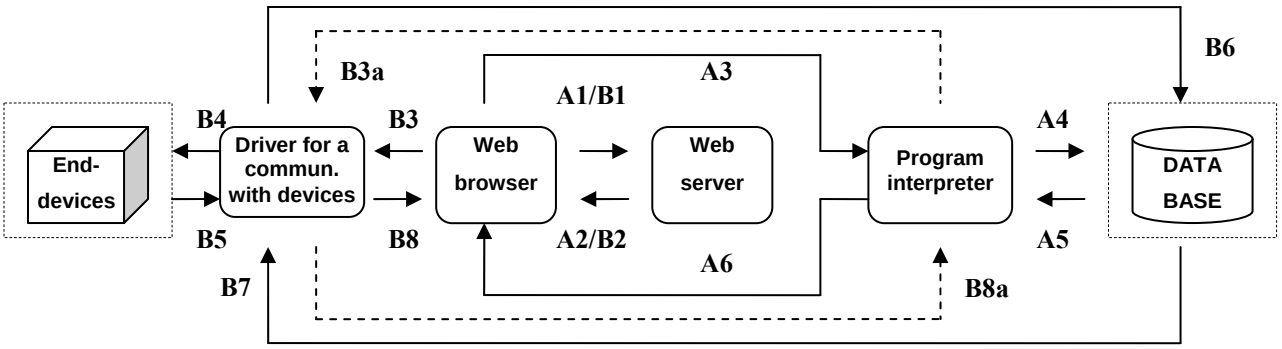

Fig. 2. Block-scheme of the architecture of the connection between a Web application and DB, where the Web-browser prepares a request for the database.

Brunch {A} completes all possible operation, {B} includes part of the operations in {A} connected with the manipulation of the data in the DB - processing, selecting, analyzing, taking decision for the control.

Branch {C} also comprises part of the operation in {A}, connected with data collecting and control. One transaction to the DB at branch {A} consists of the following steps:

- A remote station sends a request {A1} to the program module via a Web-browser, after defining some enter parameters;
- The Web-server finds the code of the program module {A2} and sends it to the program interpreter for processing and execution. The program interpreter is a part of the Web-server after it has been initialized as its module;
- The program interpreter analysis the program module. In case it contains operation for realization of a connection with the DB, as well as, searching and processing data in it, the program interpreter realizes a connection with the DB and sends {A3} to SQL requests separately;
- The DB server receives and stores the requests of SQL. It distributes them for processing and execution. In case they contain an operation whose aim is to complete operations to the ED for data collecting and control, the DB requires {A4} driver, giving it defined enter parameters. The driver sends consecutively commands {A5} to the ED. If the communication channel is not free, the SQL request has been defined as accepted and waiting for completion. An important moment at this stage is the fact that a request of SQL is able to ask a stored procedure of the DB to be executed. In this case the procedure makes possible a range of operations to be realized, which serves a definite control algorithm. This algorithm uses the data and the opportunities for processing the date in the DB and allows a great number of events that follow to be taken into consideration and solved.
- The ED receives commands from the driver. The result and the status of completing the commands {A6} are accepted back by the driver.
- The driver sends the data {A7} to the server with the DB. In this way they are stored in it to serve the program module {A8} or for a following processing, analysis or execution of a next step of the control algorithm. The request of SQL to the database is defined as processed.
- The program interpreter finishes {A9} the execution of the program module and formats the results under a code suitable for interpretation by the Web-browser;
- The web-server sends the code {A10} to the browser, and the user receives a result or information about the completion of the request with commands to an ED. In case the request is defined as accepted and waiting to be completed, the remote station is in a standby mode waiting for results.

Branch {B} in the architecture comprises part of the operations of {A} connected with the manipulation of the

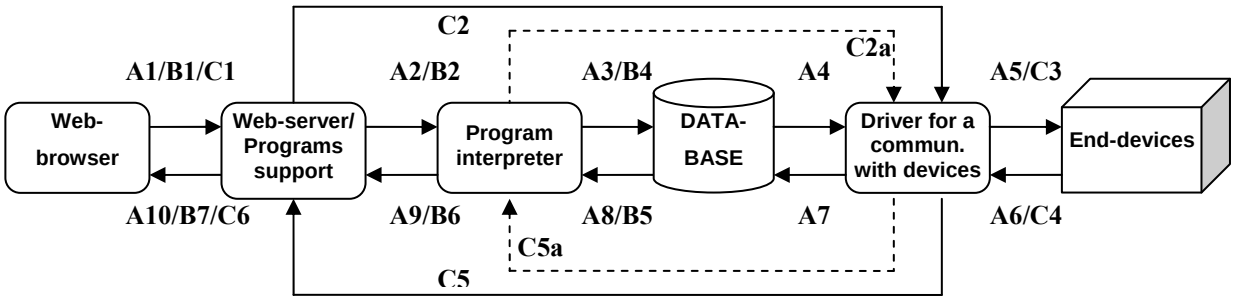

Fig. 3. Block-scheme of the extended architecture of the connection between a Web application and DB.

DB. With the help of {B} a remote station can complete operations, connected with the processing, selecting, analyzing, taking decisions for the control and projecting algorithms for control. The results from the operations, which are done here, are used by the branches {A} and {C}. Branch {C} is a limited alternative of {A}.

3. CONCLUSIONS

The architectures of the connection of a Web application with, used in manufacturing systems are presented and offered. The extended architecture which was offered presents the following advantages and opportunities:

- Analyzing and processing of data, taking decisions for the control and control by a previously drawn algorithm for control. This algorithm uses the data and the opportunities that refer to the processing of the data from the DB and it allows a great number of events that follow to be taken into consideration and solved directly by the database management system. Moreover, it allows critical in time operations to be completed, because of the possibility to systemize the processes in time.
- Distribution and control of the given to the end-devices commands by the DB.
- In this way the possibility for a remote station to give commands is limited, which could create some conflict or breakdown situations.
- Organizing of a structure which stored and sees to the execution of all the requests with commands to end-devices, as well to control their complementation. So, in a case the channel for communication with end-devices is not free, a request (which is not urgent) is stored and waiting for a next try to be completed. That makes the architecture safe and reliable.
- Succession of the opportunities of some architectures of the link of a Web-application with the database in the manufacturing systems.

REFERENCES

[1] *Microprocessor modular system for data - acquisition from analog and digital sensors*. (2003). available at: <http://msis.hit.bg/Oktopo dNet.pdf> (In Bulgarian).
[2] Stoilova, Z., *Database access trough Internet*, , available at: <http://hsi.iccs.bas.bg/Publications/ Z.I/Zl20001.pdf> (In Bulgarian).
[3] Madjarov, Ivan. (2001). *WEB and Database*, Technical University of Sofia, available at: <http://mulmedp.tu-sofia.bg/bookhtml/webdb/ pdf/webdb.pdf> (In Bulgarian).
[4] Janev, N., Ivanov, K., Karagiozov, V. (2003). *Possibility of application of Web interface at work with Database*, Proceeding of MGU "Sv. Ivan Rilski" Nr.46/3, pp. 225-231 (In Bulgarian).
[5] Rashidov, A. (2007). *Database in industrial systems*, University publisher, Gabrovo (In Bulgarian).

Author:

PhD, Aldeniz RASHIDOV, Assist. Professor, Technical University of Gabrovo, Bulgaria, Automation, Information and Control Systems Department,
E-mail: aldeniz@tugab.bg