DOI: https://doi.org/10.30837/MMP.2022.124

TECHNOLOGY OF DISTRIBUTED MANAGEMENT OF PRODUCTION PROCESSES WITH THE USE OF A SERVICE-ORIENTED APPROACH

Nevliudov I., Novoselov S.

The paper describes the technology of managing intelligent production with the use of Web services for managing distributed production processes. An analysis of the methods of implementing SOA technology in production and organization of the message exchange process was performed. The architecture of computer systems management using SOA technology and the principle of dynamic deployment of services is presented.

As a technology for the implementation of web services, it is proposed to use a distributed method of processing requests and performing services using MVC technology and the Ruby on Rails framework.

A structural diagram of the interaction of industrial equipment with the server and a generalized diagram of the organization of the web services server based on Ruby on Rails have been developed. The principle of communication between a server application and industrial equipment has been developed. The architecture of the automated production process management system using web-oriented services, taking into account the concept of Industry 4.0, was developed.

Introduction

The production process has become a collaborative activity over the years and is increasingly supported by computer systems. Thanks to joint activities, the production process has become a distributed process, where each phase of the process can be performed by different companies or individuals. This evolution became possible thanks to the development of information technologies that support the entire management process.

Currently, the production process uses several independent systems that generate a group of heterogeneous data. Managing this complex set of data involves great interoperability challenges and little flexibility, as simply changing data between systems is a challenge in itself.

Manufacturing can be understood as a set of relationships between product design, materials, production equipment and support systems, or, in short, as a set of relationships between products, processes and resources. These elements and their relationships at different levels lead to a complexity scenario, which can be seen in part as an association between understanding and managing a large body of information, which consists of three main elements: the amount of information, the variety of this information and its content, which is a measure of effort to achieve the desired result. One of the reasons for the complexity of production is the fact that data comes from different sources, generated by specific applications for each phase of product development. This leads to the need for integration between heterogeneous databases, which can be defined along three orthogonal axes: heterogeneity due to possible different interpretations; autonomy, thanks to the ability of each program to generate and manipulate data; and distribution, thanks to the location of the database components. Heterogeneous data can be managed in a variety of ways with varying degrees of distribution, flexibility, and integration.

Thus, the use of web services in production becomes more attractive because a large number of integration technologies are already used. First, Web service standards are already built into most software development tools. Second, Web services simplify engineering decisions because production applications tend to be simpler than business applications.

The relevance of research is the need to improve production systems and increase the efficiency of the functioning of industrial facilities in order to achieve high final results of activities based on the rational use of production resources.

Service-oriented approach in production management

A service-oriented approach in production management allows for the distribution of data in isolated applications, but the construction of a central model must be performed through weak links. Thus, the characteristic of autonomy of data management for each application is preserved, and the central model remains harmonious [1]. This intermediate solution not only preserves the independence of each program, but also forces each part to cooperate with the whole according to its specific capabilities. This approach allows you to insert the data of the legacy system into the central model only by creating a data adjustment interface, without the need to modify the model as a whole.

Thanks to these characteristics, this model represents the greatest flexibility without losing its characteristics of integration and distribution. The implementation of this model involves a separation layer that can request information from distributed databases, creating an interface between the partial models and the central model. A distributed data model is a solution to the integration of partial models that exist in manufacturing activities, as it meets both the requirements of distribution and integration while maintaining a high level of flexibility.

Analysis of methods of implementing SOA technology in production

The number and complexity of information systems used in companies is growing, business requirements for them are also growing, and modernizing computer-integrated systems (CIS) is becoming more and more difficult and expensive. A contradiction arises: frequent changes in business processes require IT specialists to implement them as quickly as possible, but from the point of view of economic efficiency, the cost of owning a CIS must be minimized.

Thus, for process-oriented management, a tool is needed, with the help of which it would be possible not only to effectively manage processes, but also to ensure the adaptability of CIS to changes in processes with minimal costs and in the shortest possible time. After all, one-time solutions and «patches» lead to the «hardening» of the IT solution – and, therefore, the business processes of the company, which negatively affects the overall efficiency of the business [2].

Service-oriented architecture (SOA) solves the task of increasing the flexibility of CIS, reducing the cost of software development, increasing the speed of response to changing business requirements, and also ensuring the necessary level of integration between information systems.

Figure 1 shows the control architecture of the CIS using SOA technology [3, 4].

With this approach, a general idea of the enterprise's IT architecture is created on the basis of top-level process models, which first of all implies the definition of the main types of IS that will be used. Further description of processes, at lower levels of detail, will allow to determine the main models or groups of services that will be needed to support business processes. The description of processes at the workplace level will determine the necessary functionality for services (the so-called «mapping») between business process functions and IT system services, and in the absence of the necessary service, it is necessary to formulate requirements for its development and create it.

The process organization depicted in Figure 1 implies the composition of web services to enable the execution of a complex process. This composition can be supported by BPEL. Figure 1 defines a service management architecture that has three main elements: a manager, service-oriented middleware, and a decision execution system (DSS). The manager is the main element that allows the discovery and definition of services for use by the system. In addition, it ensures the approval of service compositions. The role of the manager is to check, if the implementation of this program is possible, then the list of available services is determined. The manager deploys the management mechanism to the middleware service level.

One of the main principles of activity improvement is the reuse of previously obtained results, including software code. At one time, the repeated use of once developed functions and procedures (structural programming) was widely used.

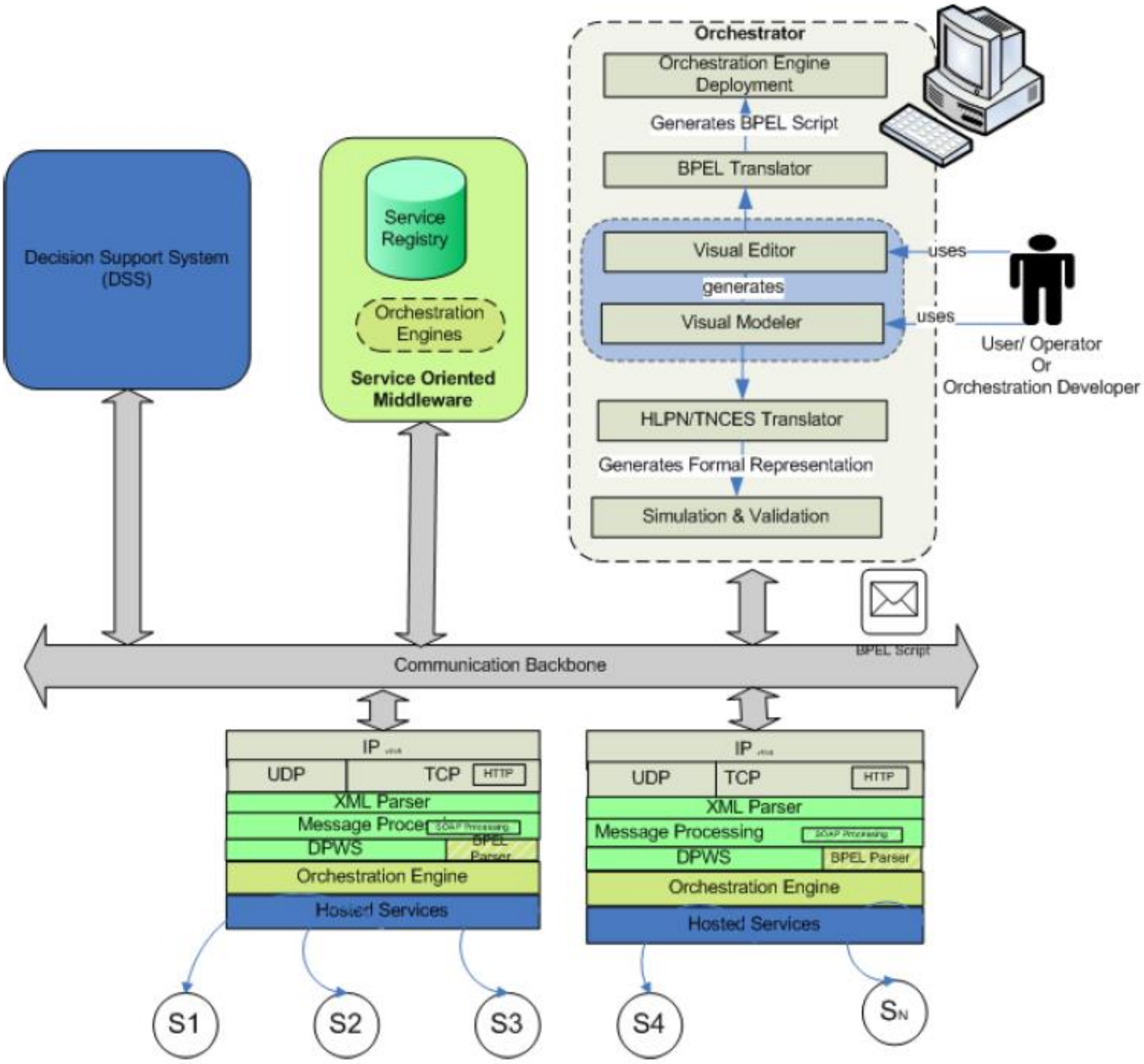

Fig. 1. CIS management architecture using SOA technology

SOA offers the principles of a process-oriented approach to building IT solutions. The developer of the IT solution formalizes the business process and connects to it typical services from the library, after which the resulting solution is transferred for execution. This approach minimizes code development. If it is necessary to make changes to the process, it is enough to change its logic without affecting the functionality of the services, which significantly speeds up the implementation of changes.

Mathematical modeling of the decentralized system

We assume that the system functions under normal conditions, therefore, the independence of individual failures can be assumed. In the proposed model, the subsystem is non-renewable.

A decentralized computing system will be understood as a set of hardware and software tools that implement the following basic functions: data processing, storage, transmission, and protection [5].

The following types of security system failures can be distinguished: hidden and false. In the event of a hidden failure of the security system, it does not respond to the failure of other subsystems, in the event of an erroneous operation, the security system involuntarily produces protective functions during the normal operation of the data storage system, data processing system, data transmission system and causes the system to stop.

The main indicator of reliability will be the probability of data loss over a certain time interval. Data loss means real destruction, data leakage, or the impossibility of accessing them for a sufficiently long period of time.

Figure 2 shows a graphic model of the functioning of RSK nodes, which reflects the failure of its subsystems and the further development of the situation.

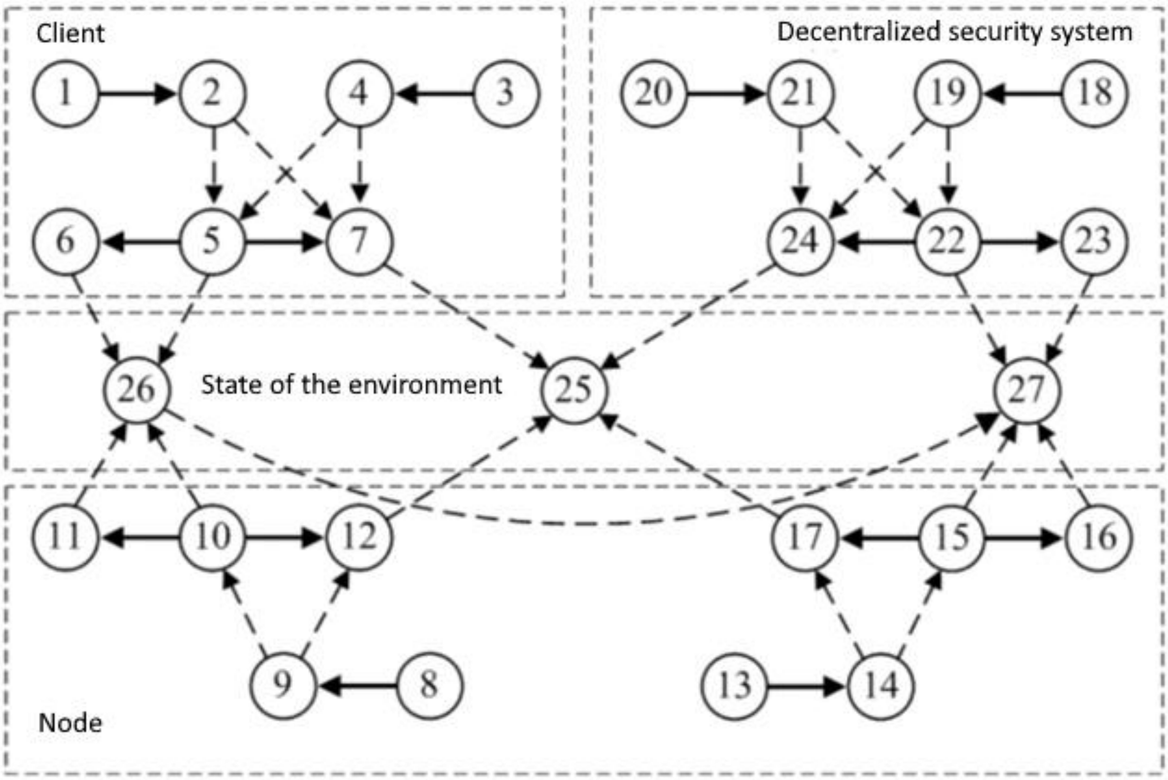

Fig. 2. Graphical model of functioning of the decentralized computing system

Solid lines indicate element transitions, dashed lines indicate the development of element failure situations. The numerical symbols in the figure mean the following:

1. Normal operation of the data processing system.
2. Data processing system failure.
3. Normal operation of the client's data transmission system.

4. Failure of the client's data transmission system.
5. Normal operation of the client's security system.
6. False failure of the client's security system.
7. Hidden failure of the client's security system.
8. Normal operation of the node of the data transmission system.
9. Failure of the node of the data transmission system to which the client is connected.
10. Normal operation of the security system node port to which the client is connected.
11. False failure of the node of the security system to which the client is connected.
12. Hidden failure of the node of the security system to which the client is connected.
13. Normal operation of the node of the data processing system.
14. Failure of a data transmission system node.
15. Normal operation of the security system node.
16. False failure of the security system node.
17. Hidden failure of the security system node.
18. Normal operation of the data storage system.
19. Failure of the data storage system.
20. Normal operation of the node of the data transmission system.
21. Failure of a data transmission system node.
22. Normal operation of the security system node.
23. False failure of the security system node.
24. Hidden failure of the security system node.
25. Decentralized computer system crash status.
26. The state of reduced efficiency of the decentralized computing system.
27. Decentralized computer system stop state.

In case of failure of the data processing system or data transmission system on the client or the failure of a node of the data transmission system to which the client is connected, with the normal operation of the corresponding security system, the client or node goes into a stop state, and the system itself goes into a state of reduced efficiency (as in the case of a false failure of a client security system or a node In the event of a storage system or data communication system failure with the corresponding client security system operating normally, the decentralized computing system enters a halt state, as well as in the event of a false failure of the security system node to which it is connected, and also in the event of the shutdown of all clients or the nodes to which the clients are connected.

If any of the subsystems fail in case of a hidden failure of the security system, the system goes into an emergency state.

We will consider the behavior of the system on the interval $[0,\ t]$. Let's enter the necessary notation:

1. Let ψ – be the failure response of a data processing system with distribution $f(t)=P(\le t)$.

2. δ – failure recovery of a distributed data storage system $f(t)=P(\le t)$.

3. $\gamma_1, \gamma_2, \gamma_3$– working time for failure of the client, nodes and nodes of the security system, respectively, having a distribution $F_1(t)=P(\gamma_1 \le t)$, $F_2(t)=P(\gamma_2 \le t)$, $F_3(t)=P(\gamma_3 \le t)$.

4. Let us denote as p_1, p_2, p_3 – the run-up to a hidden failure, and as – the run-up to η_1, η_2, η_3 a false failure of the client, nodes and nodes of the security system with the corresponding distributions:

$$F_{p1}(t)=P(p_1 \le t),$$
$$F_{p2}(t)=P(p_2 \le t),$$
$$F_{p3}(t)=P(p_3 \le t),$$
$$F_{\eta 1}(t)=P(\eta_1 \le t),$$
$$F_{\eta 2}(t)=P(\eta_2 \le t),$$
$$F_{\eta 3}(t)=P(\eta_3 \le t).$$

Reliability indicators mean the probabilities of the system transitioning into a state of stoppage and breakdown, as well as the intensity of these transitions. A shutdown of a distributed computing system will occur if all clients of the node to which the clients, security nodes, or port of the data processing node are connected enter the shutdown state.

The probability that a stop will occur in the interval $[0,\ t]$ can be written in the following form:

$$P_0(t)=1-M\left(\overline{P_{зк1}}(t)\overline{P_{зк2}}(t)\overline{P_{зк3}}(t)\right), \quad (1)$$

where $\overline{P_{зк1}}$ – the probability of stopping all customers at time t,

$\overline{P_{зк2}}$ – the probability of stopping all nodes to which clients are connected at time t,

$\overline{P_{зк1}}$ – the probability of stopping the decentralized SS at time t.

Auxiliary variables should be entered. Let ζ_i – be the time until the i-th client stops, where $i=\overline{1,N}$. Then the probability that all customers will stop in the interval $[0,\ t]$ is determined as follows:

$$P_{зк1}(t)=P\left(\mathrm{V}_{i=1}^{N}\zeta\le t\right), \tag{2}$$

where $\mathrm{V}_{i=1}^{N}\zeta_i\le t=\ \max(\zeta_1,\zeta_2,\ldots,\zeta_N)$.

Using the properties of indicators and mathematical expectation, we get:

$$\begin{aligned}P_{зк1}(t)&=MI_{\mathrm{V}_{i=1}^{N}\zeta_i\le t}=M\prod_{i=1}^{N}I_{\zeta_i\le t}=\prod_{i=1}^{N}MI_{\zeta_i\le t}=\\&=\prod_{i=1}^{N}P(\zeta_i\le t)=\prod_{i=1}^{N}\left(1-\overline{P_{ізк}}(t)\right),\end{aligned} \tag{3}$$

where $\overline{P_{ізк}}(t)$ – the probability that the i-th client will not stop at time t;

$\zeta_i\le t$ – indicator function $\left(I_{\zeta_i}\le t=1\ at\ \zeta_i\le t\ \&\ I_{\zeta_i}=0\ at\ \zeta_i>t\right)$.

The probability that the i-th client at time t will not go into the stop state:

$$\begin{aligned}&\overline{P_{ізк}}(t)=P\left(I_{p_{1i}\ge\psi_i\wedge\gamma_{1i}}\psi_i\wedge\gamma_i\wedge\eta_{1i}+I_{p_{1i}<\psi_i\wedge\gamma_{1i}}\eta_{1i}>t\right),\\&\psi_i\wedge\gamma_{1i}=\min(\psi_i,\gamma_{1i}).\end{aligned} \tag{4}$$

Development of a Web server structure for messaging in a distributed production management system

Taking into account the analysis of technologies for creating web-oriented services, it is possible to create a server structure to implement the given task.

The client communicates with the server via the HTTP protocol. The basis of this protocol is a request from the client to the server and a response from the server to the client. For requests, generally GET methods are used if we want to get data and POST if we want to publish, if we want to change data. The request is also specified in the request, the request body (if it is a request) and a lot of additional technical information [5, 6, 7].

Figure 3 shows a generalized scheme of the web services server based on Ruby on Rails.

When a request arrives at the server, it does not immediately enter the ROR. First, it is handled by the Nginx web server. If a static file (such as a file or image) is requested, Nginx itself sends it back to the client. If the request is not static, then Nginx should continue (pass) to the ROR.

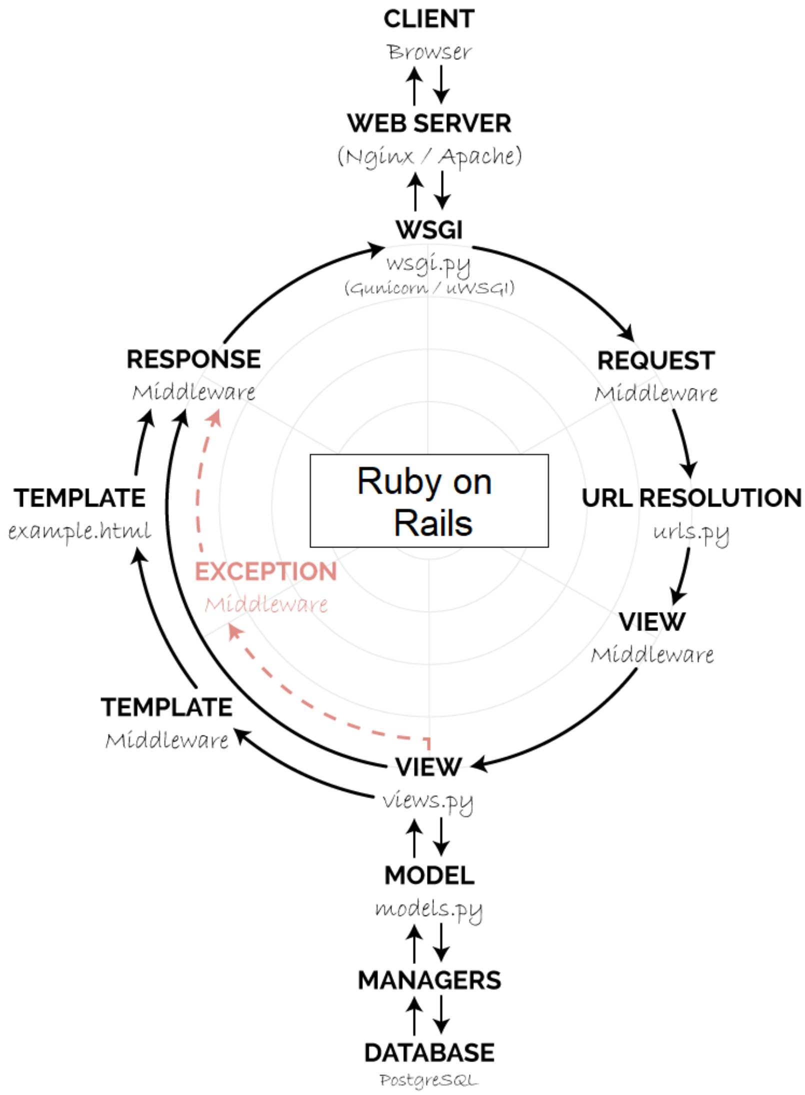

Fig. 3. Generalized scheme of the web service server organization based on Ruby on Rails

After the ROR has processed the request, it returns a response to the client in the form of HTML or a dataset and response code. If everything is fine, the response code is 200, if the page is not found, it is 404, and if an error occurred and the server could not process the request, then it is 500.

Figure 4 shows the structural diagram of the interaction of industrial equipment with the server.

In the diagram (Figure 4), industrial equipment (robot), CNC machine tool, executive device are clients of web services serving the industrial site [8]. In relation to the server, they are equal, that is, the server responds to their requests in the same way. All client devices send requests containing certain data to the server and receive responses with generated data that match the request.

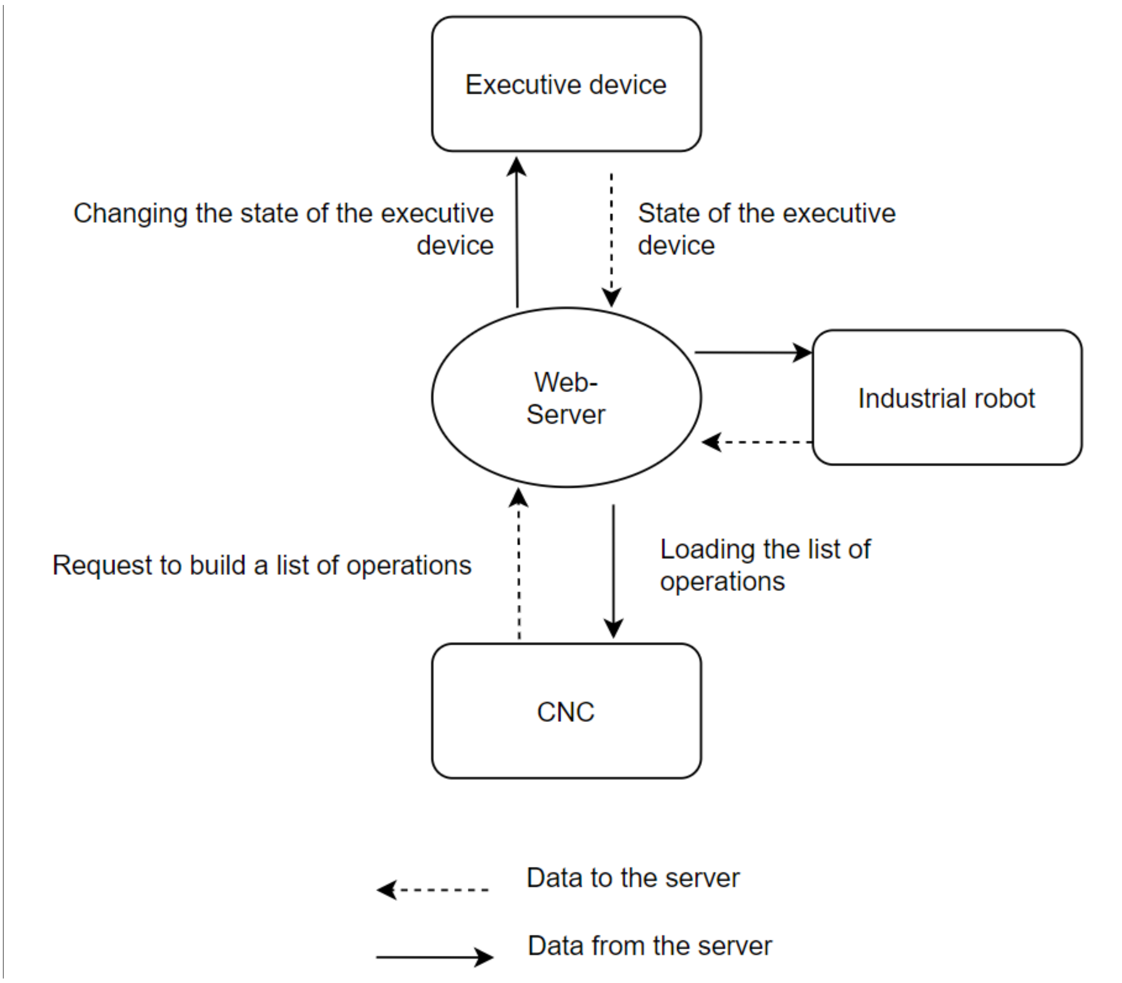

Fig. 4. Structural diagram of interaction of industrial equipment with the server

The main components that make up the server are shown in the diagram in Figure 5.

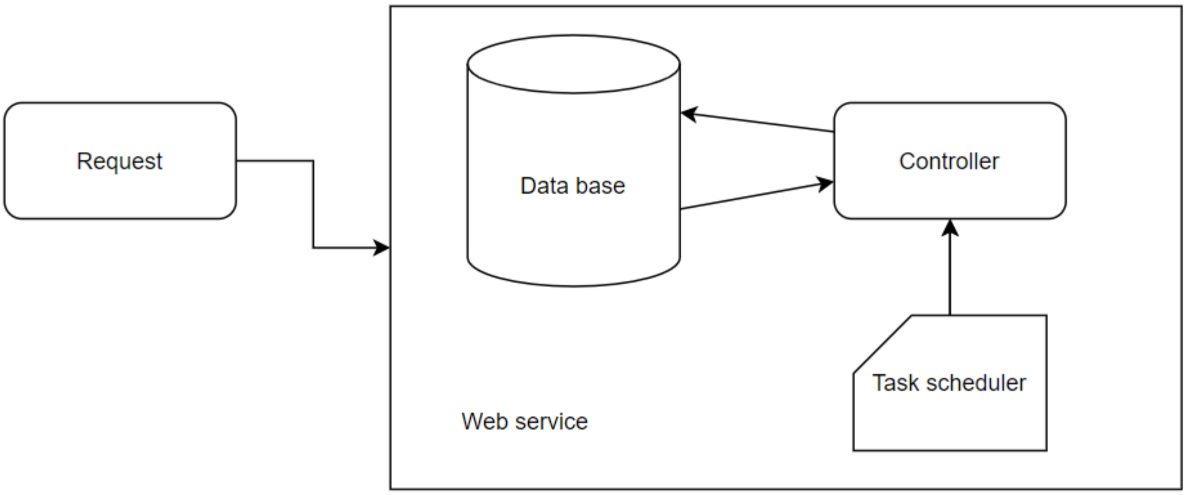

Fig. 5. The main components that make up the server

A web services server is a dedicated computer on which special software with support for the HTTP protocol is installed, with the help of which it exchanges information with its clients.

In addition, the server must have a database, task scheduler and server scripts.

Development of server software architecture

The concept of IoT, in relation to industrial automation, found its expression in the concept of Industry 4.0 [9]. «Smart factories» are characterized by the presence of cyber-physical systems: embedding computing elements (together with software and network capabilities) into physical control objects.

Taking into account this concept, the informational interaction of various types of devices and installations is an integral condition for the functioning of almost any production. This technology is called machine-to-machine interaction (M2M) and closely overlaps with IoT. In essence, IoT is a key component of the emerging Industry 4.0 concept, which involves the exchange of data between all actors involved in the production chain:

– specialists of the enterprise;

– executive components;

– ERP systems;

– robots;

– products and other systems and installations.

Let's consider the components of the production area:

– an intelligent vehicle;

– intelligent machine;

– data transfer environment;

– cloud data storage.

An intelligent vehicle has the following capabilities:

– always maintains a connection with the cloud;

– downloads the manufacturing technology of one or another part from the cloud;

– determines the trajectory of movement to the required equipment;

– checks the availability of the equipment for the next technological operation;

– moves parts between equipment;

– moves parts to temporary storage.

The intelligent machine has the following capabilities:

– a set of sensors determines the state of the main components of the equipment;

– the data transfer tool transmits information about the current state of the equipment and the stage of manufacturing the part to the cloud;

– the internal controller controls the mechanisms of the device and takes readings from the sensors;

– processes the received part according to the given technology;

– informs about the completion of the processing of the part.

Taking into account the above, the architecture of the automated production process management system using web-oriented services was built (Figure 6) [10, 11].

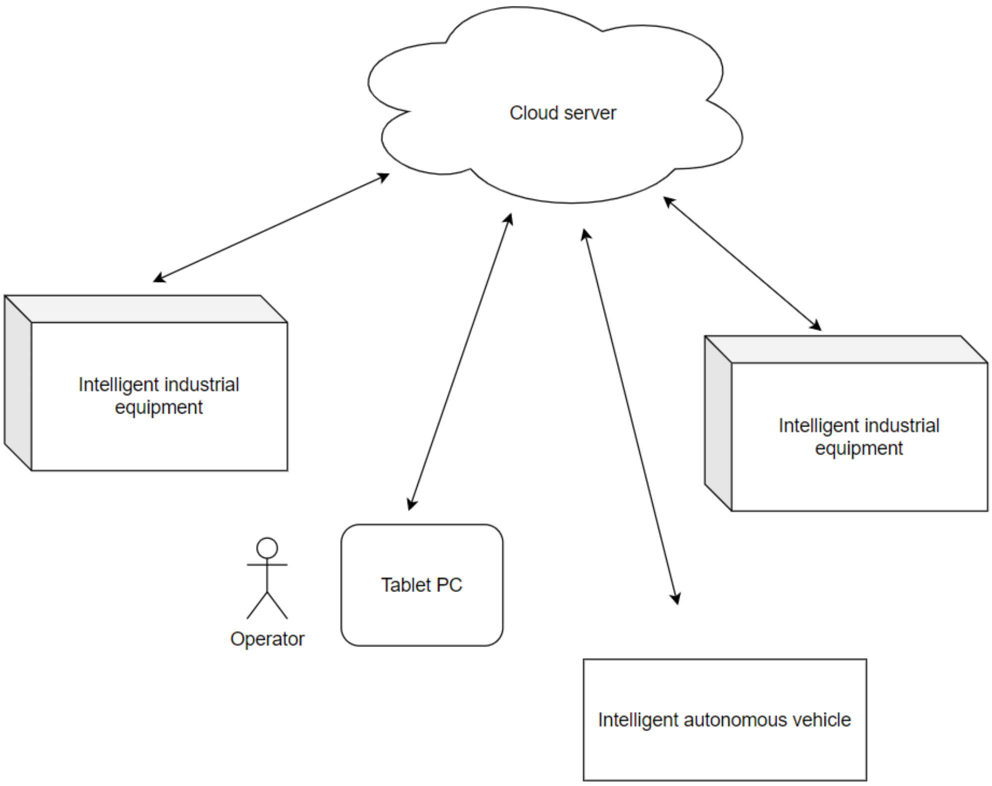

Fig. 6. Architecture of an automated production process management system using web-based services

The data transmission environment is built using wireless technology to implement the principle of mobility and scalability [12]. A simplified network protocol working on top of TCP/IP is used as a data exchange protocol for the exchange of messages between devices according to the publisher-subscriber principle.

Based on the above, we formulate the characteristics of the behavior of the production site, which is built according to the requirements of the Industry 4.0 concept:

– flexible reconfiguration of the technological process;

– easy scalability (depending on the current production load, equipment can be quickly added or removed. Each new equipment informs itself of the necessary parameters and capabilities through inter-machine communication. Depending on the received information, the behavior of the entire site changes, and intelligent transport changes the route of the delivery of blanks and transportation of semi-finished products , as well as products);

– interaction between production equipment and intelligent transport is carried out using wireless data exchange protocols;

– storage of information about equipment loading, properties of each machine, its throughput, the general state of the technological process of manufacturing parts and the current state of each individual part is stored in a public cloud on a dedicated server;

– it is possible to simultaneously manufacture several parts using different technological processes (provided the equipment nomenclature is sufficient).

To build the server, the MVC (Model – representation – controller) technology is used – an architectural template used during the design and development of software [11].

Figure 7 shows the construction architecture of the developed server.

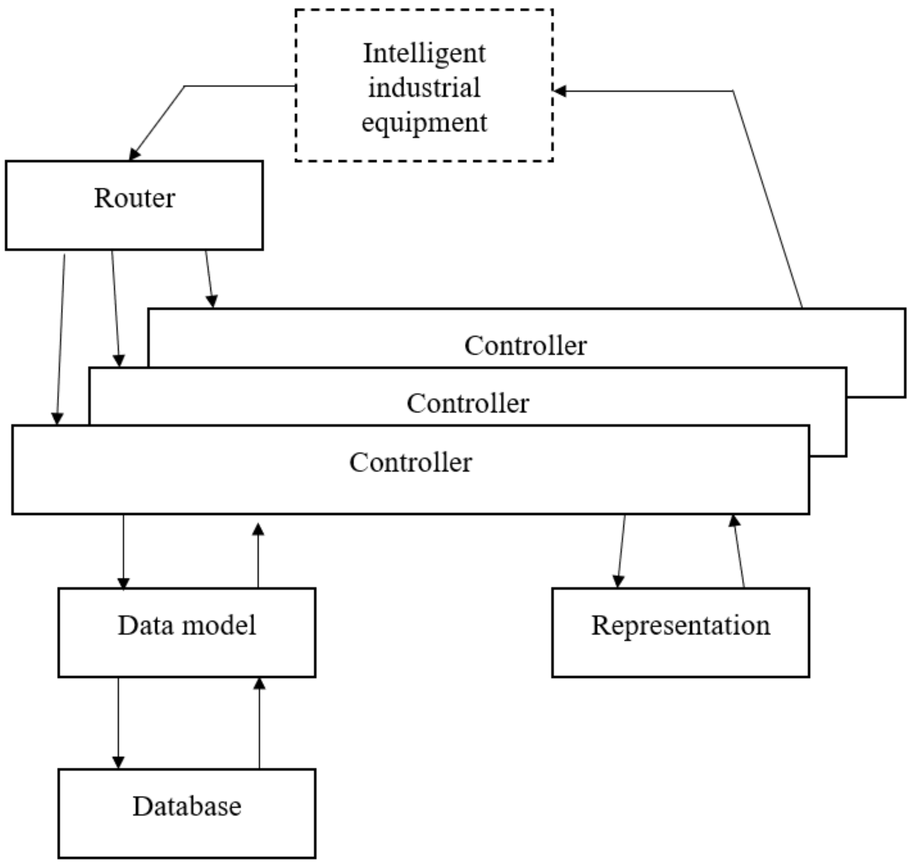

Fig. 7. Construction architecture of the server under development

Interacting with the web application of the server, the intelligent production equipment sends a request, which is accepted by the server and transmitted to the appropriate controller responsible for the chosen method. After calling the model, the controller then renders the view and returns a JSON response. By default, Rails uses the ActiveRecord library to store model objects in the relational DBMS. A competing analogue is DataMapper. The PostgreSQL relational database is used as a database. The software tool includes four controllers for processing messages from the user.

Conclusion

This work describes the technology of distributed management of production processes using a service-oriented approach. An analysis of modern technologies for deploying web services has been carried out. An analysis of the methods of implementing SOA technology in production and the organization of the message exchange process was performed. The architecture of computer systems management using SOA technology is presented. Mathematical modeling of the decentralized system was performed. As a result of the performed analysis, it is shown that the main indicator of the reliability of such a system is the probability of data loss for a certain time interval. A structural diagram of the interaction of industrial equipment with the server and a generalized diagram of the organization of the web services server based on Ruby on Rails have been developed. The principle of communication between a server application and industrial equipment has been developed. The architecture of the automated production process management system using web-oriented services, taking into account the concept of Industry 4.0, was developed.

Thus, the use of web services in production is more attractive because a large number of integration technologies are currently used and in the future even more use of intelligent devices with the use of computer-integrated control technologies is expected.

References

1. M. Hepp, «Semantic Web and Semantic Web Services», IEEE Intelligent Computing, vol. 10, no. 2, pp. 85-88, March-April 2006.
2. J. L. Martinez Lastra, I. M. Delamer, F. Ubis Lopez, Domain Ontologies for Reasoning Machines in Factory Automation, Tampere University of Technology, Institute of Production Engineering, Report 71, Tampere 2007.
3. I. Delamer, and J. L. Martinez Lastra, «Loosely-coupled automation systems using device-level SOA», In. Proc. of 5th IEEE International Conf. on Industrial Informatics (INDIN’07), vol. 2, pp. 743-747, July 2007.
4. Claudia Imhoff.Corporate information factory / Claudia Imhoff, Ryan Sousa – John Wiley & Sons, 2001 – 382 p.

5. Skakalina E. Information Technologies of Optimization of Logistic Processes / E. Skakalina // Proceeding of the 2 nd International Conference on Eurasian scientific development "East West" Association for Advanced Studies and Higher Education GmbH. – Vienna, 2014. – P. 55–63.

6. Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions / Gregor Hohpe, Bobby Woolf – Addison, 2004 – 736 p.

7. I. Nadareishvili. Microservice Architecture: Aligning Principles, Practices, and Culture / I. Nadareishvili, R. Mitra, M. McLarty, M. Amundsen – O'Reilly Media, 2016 – 146 p.

8. Klaus Schützer, Antonio Álvaro de Assis Mour; Reiner Anderl, Christian Mosch. Web service application to support distributed manufacturing // Journal of the Brazilian Society of Mechanical Sciences and Engineering. – vol. 34 no. 2 Rio de Janeiro Apr./June 2012.

9. Skakalina E. The Concept Base Model for Decision Problems of Optimal Ordering of Works / E. Skakalina // Applied Sciences and technologies in the United States and Europe: common challenges and scientific findings. Proceedings of the 7th International scientific conference. – Cibunet Publishing. New York. USA. – 2014. – P. 136–142.

10. Chris P Chris Peltz, Web Services Orchestration and Choreography, IEEE COMPUTER, October 2003. IEEE Computer Society, 2003, All rights reserved. Reprinted with permission.

11. S. Novoselov and O. Sychova, «Using Wireless Technology for Managing Distributed Industrial Automation Objects within the Concept of Industry 4.0,» 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), 2019, pp. 580–584, **DOI**: https://doi.org/10.1109/PICST47496.2019.9061333

12. Nevludov, O. Sychova, A. Andrusevich, S. Novoselov, D. Mospan and V. Mospan, «Simulation of the Sensor Network of Base Stations in a Local Positioning System in Intelligent Industries», 2020 IEEE Problems of Automated Electrodrive. Theory and Practice (PAEP), Kremenchuk, Ukraine, 2020, pp. 1–6.