Design Principles of Integrated Information Services for Batch Manufacturing Enterprise Employees

Valery Taberko,
Dzmitry Ivaniuk,
Valery Kasyanik
JSC «Savushkin Product»
Brest, Republic of Belarus
{tab,id,val}@pda.savushkin.by

Vladimir Golovko
Brest State Technical University
Brest, Republic of Belarus
gva@bstu.by

Kirill Rusetski,
Daniil Shunkevich,
Natalia Grakova
Belarusian State University
of Informatics and Radioelectronics
Minsk, Republic of Belarus
{rusetski,grakova}@bsuir.by,
shunkevichdv@gmail.com

***Abstract*—This paper discusses further applications of the ontology-based approach to the design of batch manufacturing enterprises. It involves, among other things, standards formalization. This paper, in particular, is dedicated to graphical representation of Piping and Instrumentation Diagram (P&ID) and Procedure Function Chart (PFC) languages, as per ISA-88 standard. They form a toolkit for automation engineer to work with. Contingency analysis and information retrieval agents were implemented. The article also discusses agent-oriented approach to robot interaction in robotic production systems, that is conducted via shared semantic memory.**

***Keywords*—integrated industrial control, information services, ontology-based enterprise model, Industry 4.0, cyber-physical system, ontology, knowledge base, multi-agent system, OSTIS technology.**

I. INTRODUCTION

A. Information services

One of the mainline trends of enterprise automation systems development and intellectualization involves moving away from isolated systems, each solving their own problem (CAD, SCADA, MES, ERP, WMS, SCM, CRM, etc.), to more complex integrated systems concerned with both enterprise automation and information support for clients and employees alike. Notably, the key topic of the HANNOVER MESSE 2019 [1] international exhibition is industrial intelligence within the context of integrated industry.

There are two primary lines of development for such systems: PLM (Product Lifecycle Management) systems [2] and CALS (Continuous Acquisition and Lifecycle Support) systems and technologies.

This paper pays particular attention to design principles of integrated information services for batch manufacturing enterprise employees through the JSC "Savushkin product" example. This particular approach is based on using open semantic technologies for intelligent systems. This paper uses and expands upon the results presented in [3] and [4].

Key tasks handled by such systems generally include:

- creating and maintaining shared information space at every stage of product lifecycle;
- maintaining information integrity in the shared information space during product lifecycle;
- enabling employees and automated subsystems to access, control, modify and analyse product information;
- staff training.

B. Batch manufacturing-related specifics of the information services

In regards to information service for batch manufacturing employees, as exemplified by JSC "Savushkin product", there are several basic product lifecycle stages:

1) milking;
2) shipping milk from a farm to a factory;
3) milk processing;
4) making final product;
5) bottling and packing;
6) shipping to a factory warehouse;
7) shipping from the factory warehouse to a business customer's (store, retail chain, etc.) warehouse;

Integrated information services are aimed at the following user classes, as per their production responsibilities:

- line operator controls a certain production process;
- production shop foreman manages a certain production shop;
- manufacturing director manages a certain production site;
- driver delivers raw materials and final products;
- production logistics specialist places production orders and monitors their fulfillment;

- transport logistics specialist places transportation orders and monitors their fulfillment.

C. Challenges of the information service development and approaches to facing them

There are several challenges that developers face when creating integrated systems that are aimed at solving the aforementioned problems:

- systems need to store widely heterogeneous, weakly structured information, from sensor data to rules and algorithms which specify system behaviour in case of contingency situtations;
- systems, on the one hand, should combine multiple approaches to information processing, but on the other hand, production processes are constantly evolving and thus those approaches should be constantly augmented and refined.
- opening new production facilities, structural reorganization of the enterprise, product line optimization, etc., all cause major changes to the system; such changes should require minimal effort. In other words, systems should be highly scalable and flexible.
- various devices need to interface with each other, as well as with the information system, yet they often have widely different interfaces to the outside environment;
- depending on the user category, task being performed, and other factors, there should be multiple forms of representation of the data stored in the system;

Ontology-based approach is widely used in software engineering[5], [6], as well as in other spheres[7], [8], [9], [10], to solve the problem of heterogeneous information representation. This approach involves creating a number of ontologies, at least one for each kind of information being represented in the system. A family of ontologies created for the information service being developed as part of this paper will be discussed in the following sections.

Automation industry leaders offer several solutions, which are aimed at building integrated industrial automation and servicing systems. They include "Plant EcoStruxure" from Schneider Electric [11], "MindSphere" from Siemens [12]. Such solutions have the following drawbacks:

- high entry barrier
- high cost of ownership
- their evaluation versions are limited and/or unavailable publicly
- even the developers of such systems often cannot clearly formulate future directions for their systems.

The Internet of Things (IoT) trend also concerns itself with software and hardware integration in the heterogeneous environments. Its central problems include the following [13]:

- a lot of data is gathered from various devices, but there are no tools and techniques to analyze them properly; In other words, the data can be gathered, but then cannot be processed properly;
- creating unified and standardized interfaces between devices. The matter is complicated by the fact, that the number of interfaces potentially required for direct integration of heterogeneous devices is proportional to the number of device classes, squared.
- security and access control problem.

D. Proposed approach

It is proposed to use OSTIS Technology [14] as a basis for the proposed approach to solving the problems posed. The basic principles of building a unified information system for activity enterprise automation using this technology are described by the authors in [4]. As part of this approach, an enterprise is proposed to be considered as a single information multi-agent system, within which:

- all information is combined into a single information space (enterprise knowledge base, which is stored in semantic memory);
- all participants in the process (people, robotic systems, various kinds of industrial complexes, etc.) are treated as agents over this common knowledge base. This means that they (a) monitor the situations of interest in the knowledge base and react to them (b) describe the results of their activities in the knowledge base so that this information is available to other agents and they can analyze it.
- the knowledge base of the system is hierarchically organized, i.e. represents a hierarchy of subject domains and corresponding ontologies.
- multi-agent system itself is organized hierarchically as well; agents can form arbitrarily deep hierarchies (a collective of agents can have another collective as its member). For example, several separate robotics systems can be integrated into a robotics complex to perform a particular task.

This approach has a number of advantages in regards to information services:

- no need to develop tools for direct interaction of system components (human-robotized system, human-human, etc.) due to their interaction through shared memory;
- due to the fact that all agents interact through shared memory, in the general case for the system it does not matter how this or that agent is physically arranged. Thus, the gradual replacement of manual labor with automated systems or the improvement of such systems does not require making changes to the system;

- through the use of a common single knowledge base and wide possibilities of associative search in such a knowledge base, any participant in the production process at any time has access to all the information he needs, and not to any of its predetermined fragments, the expansion of which may be related additional overhead. Thus, the various processes monitoring is greatly facilitated and the search for answers to questions of interest to users is accelerated. User requests can be refined in any way necessary;
- information stored in the knowledge base may be shown differently to different user categories; the information itself remains unchanged, only the visualization means change. Therefore, there is no need to duplicate the information.

In this work, these principles are implemented in the context of solving the problem of information services, while the main focus is on the practical implementation of these principles and demonstration of solving some of the problems discussed earlier with concrete examples.

Integrated information service development involves decomposing the system into several subsystems aimed at certain particular problems. Nowadays almost every enterprise implements some or all of these in a traditional fashion using modern technologies.

Basic manufacturing subsystems include (listed from bottom level up):

1) SCADA performs and facilitates supervisory control, manufacturing data acquisition and archival;
2) MES and WMS
    - MES (manufacturing execution system) controls product manufacturing process.
    - WMS (warehouse management) - controls product storage before shipping out to business customers.
3) ERP (enterprise resource planning) controls, which products will be produced in which quantities and with which production equipment.

That being said, every subsystem interfaces with both higher and lower level subsystems. Therefore, one of the tasks in the construction and implementation of an information service is the integration of existing subsystems into a single system and making so that these systems can be developed independently without disrupting the service as a whole. Ultimately, the construction of such a system implies a transition from the integration of heterogeneous subsystems to a single unified technological foundation for the implementation of all the aforementioned subsystems.

According to this approach, information service is considered to be an ostis-system (system that is built upon the OSTIS Technology). The traditional architecture of an ostis-system [15] includes an implementation-independent platform for interpreting semantic models of ostis-systems, and a platform-independent semantic model of this ostis-system (sc-model of an ostis-system), which, in turn, includes the following components:

- abstract semantic graphodynamic memory;
- semantic model of a knowledge base based on hierarchical system of subject domains and ontologies [16];
- semantic model of problem solver, which treats it as a hierarchical system of agents controlled by and interacting through the shared semantic memory [17]
- semantic model of an ostis-system interface (including user interface) which is treated as a kind of subsystem, that has its own knowledge base and problem solver.

[3] paper discussed higher levels of hierarchical system of subject domains and their respective ontologies for ISA-88 formalization. Further work required several additions to that system. Several ontologies were built for the following subject domains:

- Subject domain of substances
- Subject domain of products
- Subject domain of personnel
- Subject domain of manufacturing situations
- Subject domain of contingency situations
- Subject domain of logistics situations
- Subject domain of warehouse processes
- Subject domain of product shipping

Problem solver of the information service currently includes a number of basic search agents, as well as several agents that perform more complex tasks, such as identifying a reason for a certain situation. An example of such an agent is discussed below.

Information service interface consists of two parts:

- user interface provides access to the required information for various categories of end-users;
- system interface to existing enterprise automation subsystems (direct device accesss is not needed, since at this level this information can be obtained from the appropriate subsystem, e.g., a SCADA system).

The following are the examples of how the current implementation of the system interface works with both the user and other subsystems, such as SCADA.

II. SYSTEM IMPLEMENTATION

In this section, we consider specific examples of the implementation of the previously proposed principles within the various subsystems of the system being developed.

A. SCADA subsystem implementation

The main task of the information service subsystem related to the current implementation of the SCADA system is to provide interactively various reference information about the objects and concepts used in the SCADA system. This subsystem is primarily focused on servicing the foreman and shop manager.

In the current implementation, the principle of operation of the system is as follows: within the SCADA-system interface, there are interactive elements that uniquely correspond to objects and concepts in the knowledge base of the information service system (currently, connection is established through the main Russian-language identifier [18]). When a user interacts with an interactive element within the SCADA system, a request is sent to the information service system containing the identifier of the requested element, after which the system displays the semantic neighborhood of the requested element in the current state of the knowledge base.

Consider the following fragment of the "Khutorok" SCADA. Suppose, a foreman wants to get additional reference information about the current control recipe. To do this, they click the corresponding button in the project (see Fig. 1), after which the browser displays the answer to the query of the semantic neighborhood of the concept "control recipe" in SCn language (see Fig. 2).

Another way to use the information service system by the master is to identify the causes of the current situation (both standard and non-standard). In the current version of the system, the Abstract sc-agent of the search for the causes of the current state of a given object is implemented as part of the problem solver of the information service system. The specified sc-agent finds all actions in the knowledge base, as a result of which the state of the object that is the query argument has been changed.

An example of the operation of this sc-agent is given in figures 3 and 4. The valve K1Valve2 is part of the coagulator K1 and is currently open (Figure 3). After asking the question, the system, as a result of the sc-agent operation, gives an answer that the valve is open, because the washing operation is currently running for the coagulator K1 (Figure 4).

B. *Logistics subsystem implementation*

The main task of logistics subsystems is to ensure the effective interaction of the actual production, warehouse and transport.

As an example of the work of the logistics subsystem, we consider the task of monitoring the fulfillment of a production batch of products.

There are a number of stages associated with the preparation and delivery of goods batches to the customer, namely:

- batch production;
- cooling the product in a warehouse;
- loading the product on the transport;
- product delivery to the customer.

Any waiting and downtime (waiting for production, cooling the product and track downtime while loading) increase the cost of production. Thus, to minimize costs it is necessary, on the one hand, to ensure the minimum delay between these stages, on the other hand — in the event of delays quickly change the start times of the following stages.

In turn, the listed stages can be divided into more simple. For example, batch preparation consists in performing one or more recipes on production units (in accordance with S88), and recipes consist of successively performed operations. If the operation at the time of execution for some reason pauses, it means that the production time of the recipe is increased. Information about this event should go to the knowledge base of the information service system, after which they can be used to automatically or manually assess the criticality of the situation and, if necessary, adjust further stages.

For example, if the delay in the batch production process exceeds a certain amount, then it is necessary to postpone the loading time of the batch for delivery to the customer, otherwise the track will arrive and wait until the product has cooled in the warehouse. The information service system can monitor such situations and, if necessary, change the time of track departure to the warehouse for loading, taking into account the new batch readiness time.

An example of a rule in the knowledge base describing this kind of adjustment is presented in Figure 5 (it is assumed that the delay should not exceed 30 minutes).

C. *The robotic subsystem implementation*

The most important component of the information service system is the subsystem oriented to work with robotized systems. According to the above principles for building an enterprise automation system, all participants in the production process, including robots, are treated as agents working on a common information space. Thus, on the one hand, robots record the results of their activities in the general knowledge base, on the other, they can be controlled by situations and events in the knowledge base. Let us consider in more detail the use of the robotic subsystem on the example of the curd production train.

The curd production train currently includes two robotic nodes: packaging unit for the product in a box using a collaborative robot; and transportation unit for a pallet with the finished product to the receiving port of a warehouse based on a mobile robot.

Consider the task of packing curds in boxes by a collaborative robot. General view of the installation is shown in Figure 6.

The facility receives at the entrance a stream of cottage cheese, which is then divided and formed into groups of

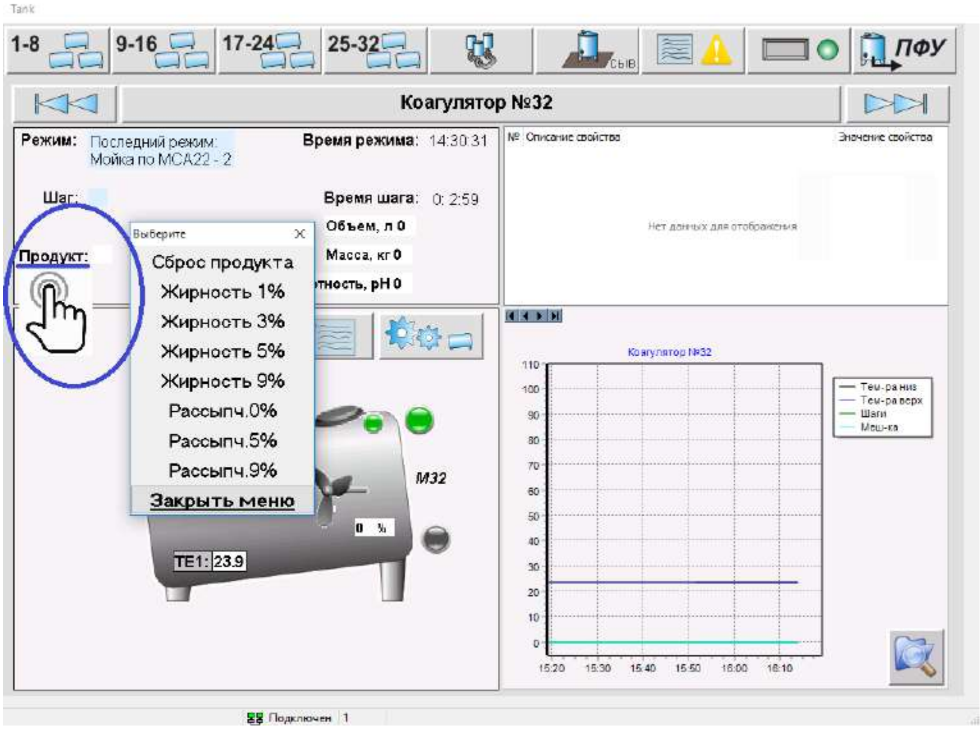
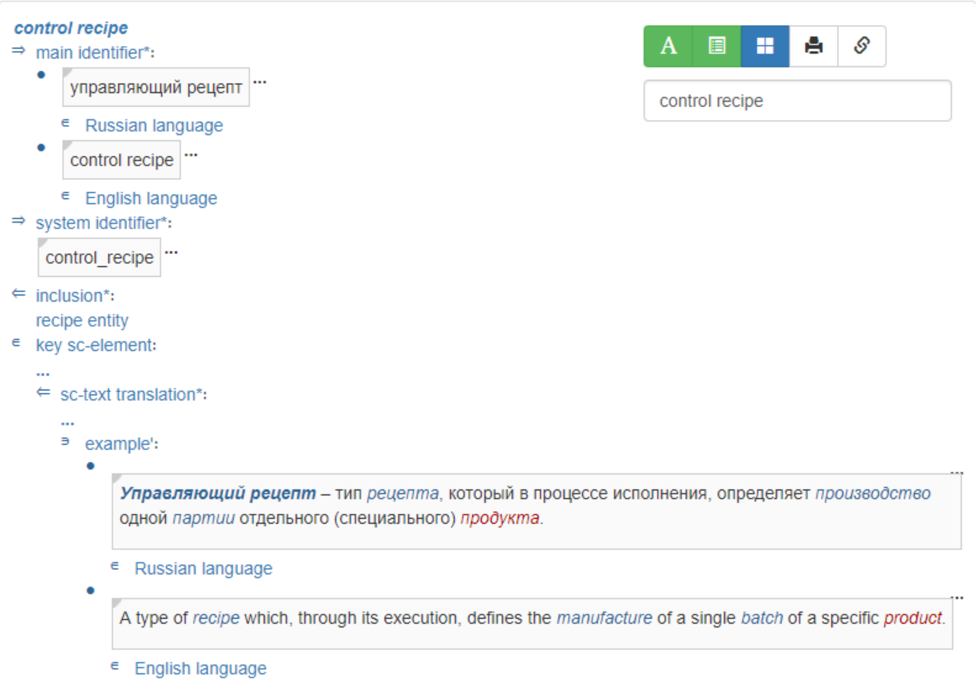


Figure 1. SCADA Request to the information service



control recipe

⇒ main identifier*:

- управляющий рецепт ...
  - ∊ Russian language
- control recipe ...
  - ∊ English language

⇒ system identifier*:

control_recipe ...

⇐ inclusion*:

recipe entity

∊ key sc-element:

...

⇐ sc-text translation*:

...

∋ example':

- ***Управляющий рецепт*** – тип *рецепта*, который в процессе исполнения, определяет *производство* одной *партии* отдельного (специального) *продукта*. ...
  - ∊ Russian language
- A type of *recipe* which, through its execution, defines the *manufacture* of a single *batch* of a specific *product*. ...
  - ∊ English language

Figure 2. Information service response

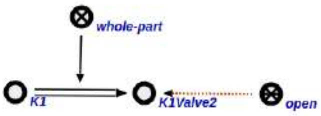


Figure 3. Relation between valve and coagulator

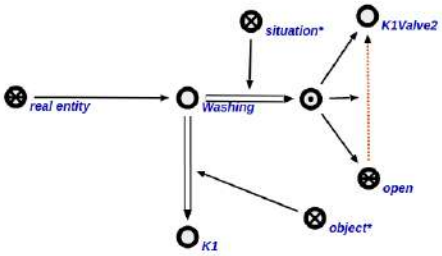


Figure 4. Valve opening causee

12 pcs. The robot with the help of a gripping mechanism takes the group and puts it in two boxes. Packing of curds is allowed in two types of boxes: single-level 6 pcs. and two-level 12 pcs. Further, the boxes on the output conveyor are issued on the palletizing unit. For the operation of the system, the robot needs the following information received from the information service system:

1) Current train performance
2) Type of box supplied
3) Lack of input boxes
4) The state of the production train nodes adjacent to the robot (Works, does not work, does not work, accident)
5) Service date of the gripping mechanism

Based on this information, the robot software flexibly responds to the current parameters of the entire manufacturing process. Adjusting the speed and delays in performing operations, the robot dynamically controls the performance of the cell so as not to create queues on the input and output conveyors. Feed box type determines the amount of product that the robot needs to pack. The state of the adjacent manufacturing process nodes allows the robot to switch the packaging process to manual processing mode or to put the product into a temporary storage device.

Next, we consider the problem of transporting the finished product to the warehouse using a mobile robot. View of the robot is shown in Figure 7.

The mobile robot operates on the concept of a mission, which consists of a product pick-up point, a delivery path and an unloading point. After starting the manufacturing process, the robot takes the loading position or expects an external signal that the pallet is full. After that, the robot starts the process of loading the pallets on board. Further, based on the available room map, the dynamic situation in the shop in compliance with safety standards, the robot moves with a pallet to the receiving port, where, if there is free space, it unloads.

The construction of the missions for the robot is performed by the information service system, for this the following subproblems are solved:

1) Creating a schedule for exporting a product from several trains, taking into account their performance and the robot;
2) Scheduling charging times during robot standby;
3) Safety level evaluation based on current situation on the production shop.

D. *User interface implementation*

The essential feature required of the information service user interface is that it should be able to visualize knowledge base information in several ways. Experienced users may choose to use universal languages, while others should be able to use more specialized languages, that are more suitable to their line of work. As system functionality grows, the need to expand the number of external languages used may arise.

To be able to expand the number of external languages used, OSTIS technology-based computer systems need the appropriate interface tools. An approach to developing such tools was proposed in the [19] paper. This approach requires three ontology-based models for each external language: semantic model of language texts, syntactic model of language texts, and semantic-syntactic transformation model in the form of ontology of transformation rules.

Current version of the information service uses two external visualization languages. Procedural models are visualized using PFC language, and physical models are visualized using P&ID language. [4] provided several examples of formal representation of ontology-based models for PFC language. PFC and P&ID visualization tools have been introduced in the current version of information service system. Figures 8 and 9 show images produced by the current visualization tools.

Furthermore, to simplify interaction for inexperienced users, the simplified visualization mode was developed. It hides parts of the information needed only for the knowledge base developers. Visualization mode, which shows all the information, is called an "expert mode", and can be switched on or off with a toggle. Figures 10 and 11 show representations of the "procedure" term specification, as displayed in simplified and expert modes.

III. CONCLUSION

The paper discusses the principles of construction and the current version of the system of information services for an batch manufacturing enterprise on the example of JSC «Savushkin Product».

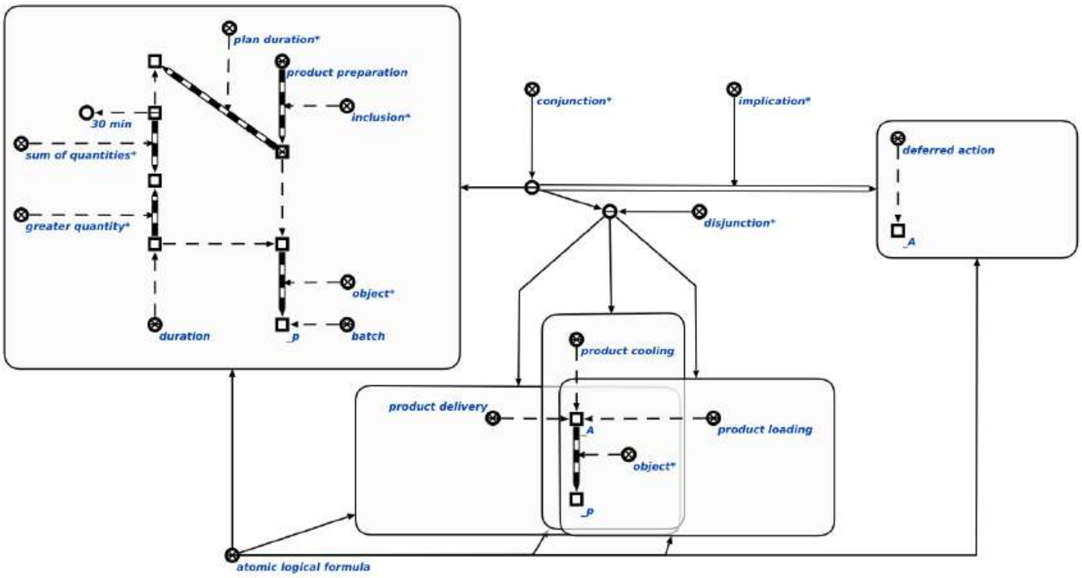


Figure 5. Rule describing the adjustment in case of delay in production

Figure 6. Curds laying module

Figure 7. Mobile Transport node

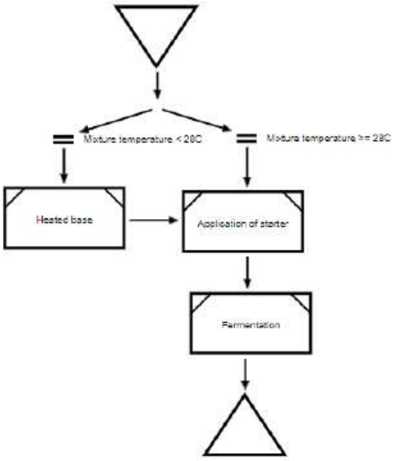


Figure 8. PFC visualization example

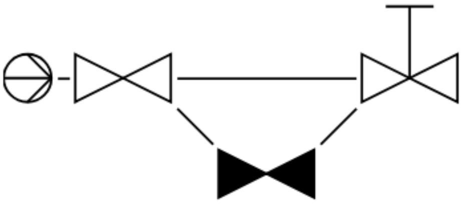


Figure 9. P&ID visualization example

Development plans of the considered system include the following urgent tasks:

1) Refusal to edit the source code of the knowledge base in favor of editing the knowledge base in real time, which will make the work of users more efficient and allow employees of the enterprise (for example, automation engineer) to make changes to the knowledge base independently.
2) Graphical representation modernization: in the engineering documentation, every little thing matters (thickness, color and kinks of the line), in the current implementation the display is somewhat simplified.
3) Tighter integration of already implemented systems with the information service system, building a unified system that combines several levels of the enterprise.

ACKNOWLEDGMENT

The authors would like to thank the staff and administration of the JSC "Savushkin Product" for their support of the work.

REFERENCES

[1] (2018, Dec.) Integrated industry - industrial intelligence. [Online]. Available: https://www.hannovermesse.de/en/news/key-topics/integrated-industry/index-2.xhtml

[2] J. Stark, *Product Lifecycle Management (Volume 1): 21st Century Paradigm for Product Realisation*, 3rd ed. Switzerland: Springer, Cham, 2015.

[3] V. Taberko, D. Ivanyuk, K. Rusetski, D. Shunkevich, I. Davydenko, V. Zakharov, V. Ivashenko, and D. Koronchik, “Ontology-based design of batch manufacturing enterprises,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2017, pp. 265–280.

[4] V. Taberko, D. Ivaniuk, V. Kasyanik, V. Golovko, N. Guliakina, K. Rusetski, D. Shunkevich, A. Boriskin, and N. Grakova, “Design of batch manufacturing enterprises in the context of industry 4.0,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 307–320.

[5] T. Dillon, E. Chang, and P. Wongthongtham, “Ontology-based software engineering- software engineering 2.0.” *Australian Software Engineering Conference, IEEE Computer Society*, pp. 13–23, 2008.

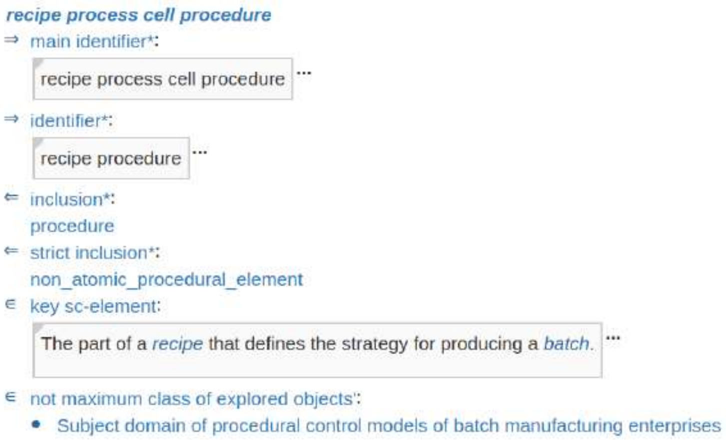
[6] A. Emdad, “Use of ontologies in software engineering,” 2008, pp. 145–150.

[7] A. N. Andrichenko, “Tendencii i sostojanie v oblasti upravlenija spravochnymi dannymi v mashinostroenii,” *Ontologija proektirovanija*, no. 2(4), pp. 25–35, 2012.

[8] A. Fedotova and I. Davydenko, “Primenenie semanticheskih tehnologij dlja proektirovanija intellektual’nyh sistem upravlenija zhiznennym ciklom produkcii,” *Izvestija vysshih uchebnyh zavedenij. Mashinostroenie*, no. 3 (672), pp. 74–81, 2016.

[9] A. Fedotova, I. Davydenko, and A. Pfortner, “Design intelligent lifecycle management systems based on applying of semantic technologies,” vol. 1. Springer International Publishing Switzerland, 2016, pp. 251–260.

[10] A. Fedotova, V. Tarassov, D. Mouromtsev, and I. Davydenko, “Lifecycle ontologies: Background and state-of-the-art.” Copenhagen: IARIA XPS Press, 2016, pp. 76–82.



recipe process cell procedure

⇒ main identifier*:

recipe process cell procedure ...

⇒ identifier*:

recipe procedure ...

⇐ inclusion*:

procedure

⇐ strict inclusion*:

non_atomic_procedural_element

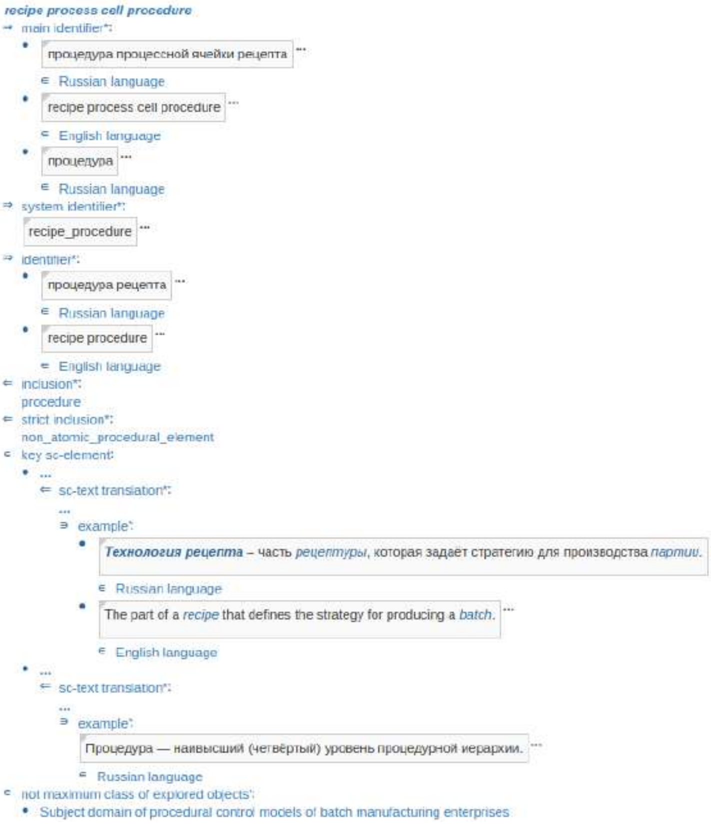
∈ key sc-element:

The part of a *recipe* that defines the strategy for producing a *batch*. ...

∈ not maximum class of explored objects':

- Subject domain of procedural control models of batch manufacturing enterprises

Figure 10. Simplified representation



recipe process cell procedure

⇒ main identifier*:
- процедура процессной ячейки рецепта ...
  - ∈ Russian language
- recipe process cell procedure ...
  - ∈ English language
- процедура ...
  - ∈ Russian language

⇒ system identifier*:

recipe_procedure ...

⇒ identifier*:
- процедура рецепта ...
  - ∈ Russian language
- recipe procedure ...
  - ∈ English language

⇐ inclusion*:

procedure

⇐ strict inclusion*:

non_atomic_procedural_element

∈ key sc-element:
- ...
  - ⇐ sc-text translation*:
    - ...
    - ∋ example':
      - ***Технология рецепта*** – часть *рецептуры*, которая задаёт стратегию для производства *партии*. ...
        - ∈ Russian language
      - The part of a *recipe* that defines the strategy for producing a *batch*. ...
        - ∈ English language
- ...
  - ⇐ sc-text translation*:
    - ...
    - ∋ example':
      - Процедура — наивысший (четвёртый) уровень процедурной иерархии. ...
        - ∈ Russian language

∈ not maximum class of explored objects':
- Subject domain of procedural control models of batch manufacturing enterprises

Figure 11. Expert mode representation

[11] (2018, Dec.) Ecostruxure is iot-enabled, plug-and-play, open, interoperable architecture and platform, in homes, buildings, data centers, infrastructure and industries. [Online]. Available: https://www.schneider-electric.ru/ru/work/campaign/future-of-automation/smart-industrial-automation.jsp

[12] (2018, Dec.) Mindsphere is the cloud-based, open iot operating system from siemens that connects your products, plants, systems, and machines, enabling you to harness the wealth of data generated by the internet of things (iot) with advanced analytics. [Online]. Available: https://www.siemens.com/global/en/home/products/software/mindsphere.html

[13] (2018, Dec.) 7 big problems with the internet of things. [Online]. Available: https://www.cmswire.com/cms/internet-of-things/7-big-problems-with-the-internet-of-things-024571.php

[14] V. V. Golenkov and N. A. Guljakina, “Proekt otkrytoj semanticheskoj tehnologii komponentnogo proektirovanija intellektual’nyh sistem. chast’ 1: Principy sozdanija,” *Ontologija proektirovanija*, no. 1, pp. 42–64, 2014.

[15] V. Golenkov, N. Guliakina, N. Grakova, I. Davydenko, V. Nikulenka, A. Eremeev, and V. Tarassov, “From training intelligent systems to training their development tools,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 81–98.

[16] I. Davydenko, “Semantic models, method and tools of knowledge bases coordinated development based on reusable components,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 99–118.

[17] D. Shunkevich, “Agent-oriented models, method and tools of compatible problem solvers development for intelligent systems,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 119–132.

[18] (2018, Nov.) IMS metasystem. [Online]. Available: http://ims.ostis.net/

[19] A. Boriskin, D. Koronchik, M. Sadowski, I. Zhukov, and A. Khusainov, “Ontology-based design of user interfaces,” in *Otkrytye semanticheskie tehnologii proektirovanija intellektual’nyh sistem [Open semantic technologies for intelligent systems]*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2017, pp. 265–280.

ПРИНЦИПЫ ПОСТРОЕНИЯ СИСТЕМЫ КОМПЛЕКСНОГО ИНФОРМАЦИОННОГО ОБСЛУЖИВАНИЯ СОТРУДНИКОВ ПРЕДПРИЯТИЯ РЕЦЕПТУРНОГО ПРОИЗВОДСТВА

Таберко В.В., Иванюк Д.С.,
Касьяник В.В., Головко В.А.,
Русецкий К.В., Шункевич Д.В., Гракова Н.В.

В данной работе предлагается продолжение применения онтологического подхода к проектированию предприятий рецептурного производства. В рамках продолжения формализации стандартов рассматривается графические представления внешнего языка спецификации процедурных моделей Procedure Function Chart и P&ID-схем (стандарт ISA-88), дающие инженерам привычный инструмент для работы. Реализованы агенты получения дополнительной информации и анализа типовой нештатной ситуации. Рассматривается также агентно-ориентированный подход к организации взаимодействия роботов в рамках роботизированных производственных комплексов, основанный на взаимодействии через общую семантическую память.

Received 10.01.19