Characteristic and comparison of UML, BPMN and EPC based on process models of a training company

Marcin Nizioł, Piotr Wiśniewski, Krzysztof Kluza and Antoni Ligęza
AGH University of Science and Technology
al. A. Mickiewicza 30, 30-059 Krakow
Email: {wpiotr,kluza,ligeza}@agh.edu.pl

Abstract—We describe, characterize and compare three selected modeling notations of business processes: Unified Modeling Language, Business Process Model and Notation, as well as Event-Driven Process Chain. Using processes implemented in a training company, the selected notations were discussed in detail. We compare various aspects, such as modeling notation origin, the number of graphical elements included. Moreover, notations were analyzed using the 4+1 architectural view model. Justified results of the survey conducted among employees of above-mentioned organization let us conclude that there exist notation differences. Both BPMN and EPC allow the process architects to prepare more precise and legible models than UML.

Index Terms—Business Process Management, BPMN, EPC, UML, process modeling

I. Introduction

BUSINESS process modeling is a graphical representation of processes taking place in organizations. Process models are most often developed by process analysts. They present how the organization and its structures work. At the same time, they provide information that helps define the way the organization should act and indicates the direction of change. Such models may be further automated, as growing interest in the robotic process automation might be observed [1].

Modeling of business processes allows observation of their implementation, and thus the optimization of processes (simplification, increased transparency) or duration. It indicates which employee will be responsible for implementation at a given stage, and also allows to determine who is responsible for a given fragment of the implemented process. The most popular notation of business process modeling is Business Process Model and Notation (BPMN) while Unified Modeling Language (UML) or Event-Driven Process Chain (EPC) are also used successfully.

In this paper, the aforementioned notations have been thoroughly characterized with the help of original drawings, which present graphic elements, fragments or models of the entire processes of a training company.

The paper is organized as follows: Section II presents basic information about the UML language. Section III characterizes the BPMN notation while Section IV describes the EPC language together with the ARIS methodology [2]. We also compare these three ways of modeling business processes (Section V). The list is based on existing sources and proprietary models. The two processes of the above-mentioned organization are presented graphically – each in three variants. This section also presents the results of the survey of employees of an enterprise dealing with the organization of training. The last section (numbered as Section VIII) presents a summary containing the overall conclusions of the conducted comparison and presents ideas for future work.

II. Unified Modeling Language (UML)

The Unified Modeling Language (UML) [3] from the Object Management Group (OMG) is a standardized notation for modeling object-oriented software applications [4]. This multipurpose modeling language offers a variety of notations to capture different aspects of software [5], [6]. UML has become the dominant notation among software engineers and attempts to be a universal visual notation for software design.

UML is a quite complex notations, which makes it hard to understand by non-technicians [7] and is not suitable for all aspects of modelling [8]. Although it was created for modeling IT systems and is constantly developed in this area, it can be successfully used as a notation of business process modeling [9]–[14].

Due to the fact that UML is very popular and widely known, we decided to use it also at the business level. An analogy was also noted between an IT system and a business process – both are modeled from two perspectives – the structure and course of the process (dynamic structure). However, it should be remembered that Unified Modeling Language is not dedicated to business processes, and using it for this purpose may carry some risk – for example, ambiguous interpretation of the model caused by inconsistent understanding of the presented elements in the context of the created business process [15].

Although the UML offers over a dozen of diagram types, in the case of business process models, activity diagrams are used the most often [16]. A UML activity diagram is responsible for presenting the system dynamics. This is the type that is used to prepare material for business analysts in organizations. Activity diagrams are also used in modeling systems, algorithms or use case scenarios. The following graphic elements are used to construct these types of diagrams:

- **Activity** – behavior of the actor of the modeled process. Presented as a rectangle with rounded edges. One activity can consist of more than one subactivity. Action names are formulated in imperative mode, most often placed inside the element;
- **Action** – specifies the activity. The graphic notation of this element is the same as for activities;
- **Control Flow** – represents the relationship between actions and activities, as well as the sequence of flow between them. Represented by an arrow;
- **Start node** – the point that initiates the start of the process. Represented as a filled black circle. Most often there is one beginning for one diagram, while there may be more in modeling complex processes and systems;
- **End node** – the point where the process ends. In the diagram it will be an empty circle with a black dot inside. Activity diagrams may appear more than once;
- **Flow End** – the moment at which the selected control flow is stopped. It can occur repeatedly. Presented using unfilled crossed with two lines circle;
- **Decision node** – represented by a diamond. This is the place where the decision is made determining further control flows. A logical condition is placed next to the decision block (in square brackets, in the form of an infinitive). The number of outgoing flows from the block depends on the number of results of the logical condition. These also receive their names. For the decision block to make sense, the output results must be mutually exclusive.

Figure 1 shows an example UML activity diagram prepared for the administration department of the training company. The diagram represents a fragment of the large onboarding process of a new employee who needs to have a valid certificate after training in OHS rules to start working in a new position. The model consists of a small number of elements. The first step is checking by an employee of the administration department whether the person being onboarded does not have valid documents. The flow then goes into the decision block (*Valid Certificate*), from which two control flows are outgoing. One of them (*Yes*) leads into the graphic element *end*. The new employee has provided current documents, organization of OHS training will not be necessary. The other result of the decision block (*No*) directs the flow to the next step followed by *end*, where the process ends.

III. BUSINESS PROCESS MODEL AND NOTATION (BPMN)

Business Process Model and Notation [17], [18] was created and provided by the Business Process Management Initiative (BPMI) [19]. Its current version – 2.0.2 (introduced in January 2012) and the standard are maintained by the Object Management Group (OMG). The goal that guided the creators of BPMN notation was to create a language to describe processes taking place in the enterprise that would be understandable for all business users. The notation was to be universal and unambiguous enough to make graphic representations legible
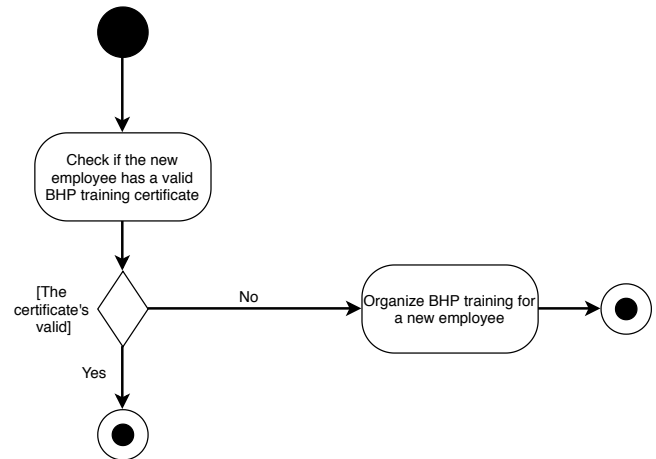

Figure 1. Example UML activity diagram.

and understandable for analysts responsible for creating models, technical persons dealing with the implementation of processes, not forgetting about the business representatives who manage and monitor workflows [20], [21]. Preparation of such models may also support process optimization, which usually has a positive impact on the efficiency of the organization and the time users spend on getting familiar with processes.

A. BPMN Process Diagrams

Process diagrams constitute the main type of models within the BPMN notation. The basic graphical elements of a process diagram are [20], [21]:

- activities;
- events;
- logical gateways;
- sequence flows;
- message flows
- pools and lanes;
- data objects;
- artifacts.

The BPMN standard alone does not specify the level of detail in modeling. This means that not all graphic elements need to be used in the final process model. It is its intended use that determines how accurate the prepared diagram should be. Drejewicz [20] lists three levels of detail of a model prepared using the BPMN notation:

1) **Illustrative model** – intended to present only general assumptions in the process. In this case, there is no description of technical issues, penetration into details of flows, nor the presentation of subprocesses.
2) **Analytical model** – prepared for the purpose of analyzing tasks that will be performed when creating and implementing the process in an organization. In this case, attention is paid to the use of data types, subprocesses, flow types, gateways and tasks.
3) **Executable model** – the most detailed business process model. It should include as much information as possible about the implemented business process.

Figure 2 illustrates an example model of the payment process in the training company. It consists of a small number of activities (represented by rounded rectangles): the actor of the process chooses the payment method, performs the payment. Then, the flow goes to the logical gateway (diamond), which indicates that two results of the decision are possible – the payment is successful or, if the actor does not have enough funds on the account, the payment is rejected. Arrows represent sequence flows – the order of actions performed in the process, i.e. the priority of activity execution, as well as time dependencies [22]. The line that connects the logical gateway to the activity *Realize Payment* has an additional cross section – this line determines the default sequence flow. The circle with a thin edge and the bold circles represent events – the start event and the end event, respectively. Figure 2, due to the very small amount of details presented and the lack of decomposition into individual actions can be considered an illustrative model.
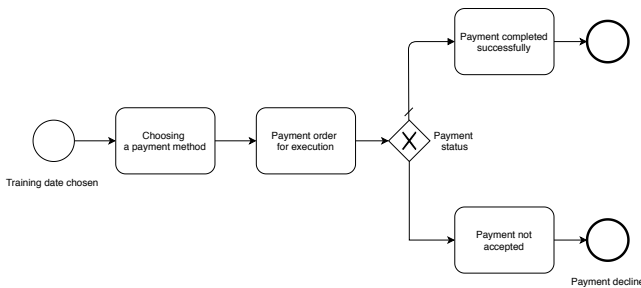

Figure 2. Example BPMN process model.

B. BPMN Choreography Diagrams

An interaction of two processes or two major participants in a process, can be represented using BPMN in the form of a collaboration diagram or a choreography. A BPMN collaboration diagram is, in fact, a combination of two or more pools with message flows between them.

Figure 3 shows an example collaboration diagram of confirming the customer's enrollment for an open training. There are two process participants in this case – a customer and a customer service employee, which is why there are two pools in the model that communicate by exchanging messages.
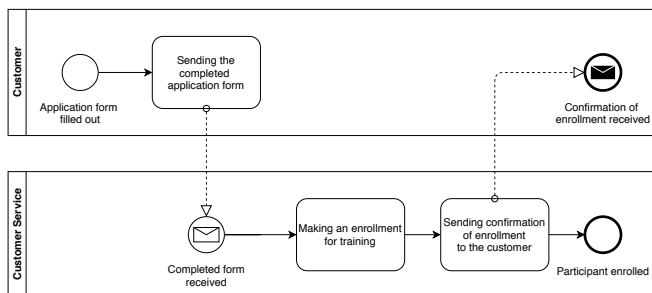

Figure 3. Example BPMN collaboration diagram of registration confirmation.

A choreography is a kind of process, but it differs significantly in purpose and behavior from a standard BPMN process model, which usually presents step-by-step activities. Choreographies focus rather on the ordered flow of information and the interactions between two participants of the process or two processes. The difference is also that in a standard process model, we can present the actions of one major actor, and choreography requires the presence of at least two. Therefore, it is impossible not to notice the very important relationship between choreography and BPMN pools. Since a pool is a graphic representation of one participant in the process, choreographies will take place only between pools. The following graphical elements are used in choreography diagrams:

- choreography activity – presented using a rectangle with rounded corners divided into three parts. The sender and recipients of the message are placed in the upper and lower part. It does not matter which actor is in which part, but the section representing message recipients is filled with a dark background, e.g. gray. The middle part contains the name of the activity being carried out;
- complex choreographies - a type of task that consists of various choreography tasks. It may also appear as several exchanges of messages between process actors;
- events;
- sequence flows;
- logical gateways.

Figure 4 shows the corresponding choreography diagram for this process. Two participants take part in the process, which is why only two actors appear in the choreography.
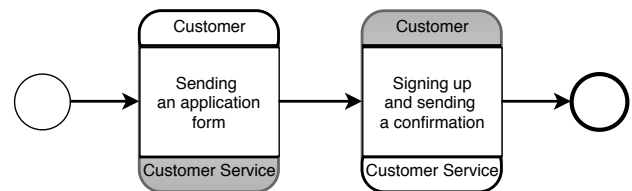

Figure 4. Example BPMN choreography diagram of registration confirmation.

IV. EVENT-DRIVEN PROCESS CHAIN (EPC)

Event-Driven Process Chain is another notation that is used in modeling, redesigning facilitating of business processes, as well as controlling and organizing workflows. EPC was provided as part of the work on the ARIS method by August-Wilhelm Scheer from the University of Saarland in the early 1990s [23]. A model prepared using EPC is an ordered diagram of events and functions, combined flows and logical operators: OR, XOR or AND [24]. Additional passive elements, such as documents, systems, tools and data objects, can be used to refine the model [25].

The biggest advantage of the Event-Driven Process Chain language is its simplicity and intuitiveness [26]. What is more, the syntax does not include too many graphic elements, and thus easy to interpret. There were also attempts to formalize

the semantics of EPC [27], [28]. Although EPC is considered an informal notation, analysts successfully use it to prepare professional and detailed business process models [29].

The diagram presented in Figure 5 is an example of a business process model prepared using the EPC language.
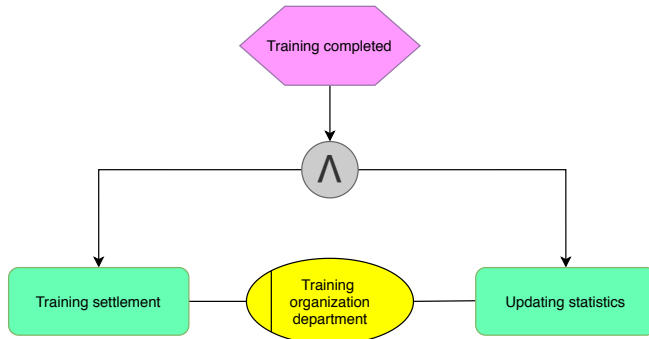

Figure 5. Example EPC diagram.

V. COMPARISON OF NOTATIONS USING THE PROCESS MODELS OF A TRAINING COMPANY

Event-driven Process Chain [30] was the first to be released. Work on the original version of Unified Modeling Language began four years later, in 1994 [15]. However, the Business Process Model and Notation language (currently the most popular notation of business process modeling) is the youngest – its first version was released in 2004 [20]. Over the years, managing organizations have developed and updated them – for example, UML has been modified more than 15 times, and BPMN notation has gained 4 subsequent versions since 2004 – there are 5 in total, the current version 2.0.2 was released in 2014 [21]. The EPC language has not changed since its introduction. BPMN and EPC were created and developed for convergent purposes. First of all, they are to enable graphical representation of processes that take place in organizations for the stakeholders taking part in their implementation. Unified Model Language is a notation dedicated to creating models of information systems, used in software engineering. The multitude of diagrams that it offers allows the modeler to present a complete IT system in a view of many models. However, this does not preclude using UML to create business process diagrams – this is successfully practiced.

A. Comparison of graphical elements

Unified Model Language, Business Process Model and Notation, as well as Event-driven Process Chain have a number of different graphical elements from which business process models are built. These elements, although different in appearance, name or adopted rules of use, play convergent roles in the models.

B. Comparison based on Kruchten's 4+1 view model

"4 + 1" [31] is a view model presented by Philippe Kruchten, used to compare views of system specifications and a description of software architecture. Using this tool,

it is possible to analyze an IT system - from five concurrent views, each of which deals with a different set of issues. These views present the perspectives of different users of the created software (business, suppliers or end users) [16]. The views included in the 4+1 model are:

1) Logical view – describes the object model of the process, occurs at the conceptual level.
2) Process view – presents aspects of concurrency and process synchronization, also applies to the conceptual level.
3) Development view – describes static organization of software in a development environment [31].
4) Physical view – presents software mapping on hardware.
5) Use case view – presents usage scenarios of the system.
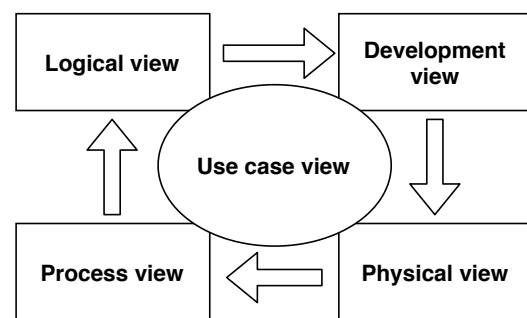
Figure 6 shows a 4+1 view model architecture.

Figure 6. The 4+1 view model.

The first four views are used to to register design decisions, the fifth allows the user to illustrate them and then verify [31]. Figure 7 compares the use of UML (activity diagram), BPMN (process and choreography) and EPC notation using the "4 + 1" view model. A filled diamond means that the diagram is used in the particular view, while a partially filled diamond stands for the possibility to use a certain diagram in the particular view. As it can be seen in the figure, process modeling notations are not present in the physical view.

C. Comparison of process models used in the training company

The models used for our research represent two business processes implemented in a mid-sized training company from Krakow, Poland. This company deals with the sale and organization of open and closed training in project, portfolio, risk and change management standards etc. The first process describes the confirmation of an open training, in which customer service employees take part. Figure 8 presents this process model in the EPC notation.

The second process presents actions taken in the workflow of booking a trainer, performed by customer service or representatives of the sales department, sales director and the trainer himself. Figure 9 presents this process model in the EPC notation.

For our analysis, we used diagrams in UML (version 2.0), BPMN (version 2.0) and EPC notations.

	UML Activity Diagram	BPMN Diagram	BPMN Choreography Diagram	EPC Diagram
Logical view	◇	◇	◇	◇
Process view	◆	◆	◆	◆
Development view	◇	◈	◈	◇
Physical view	◇	◇	◇	◇
Use case view	◈	◈	◇	◈

Figure 7. Comparison of different process representation in terms of the 4+1 view model.
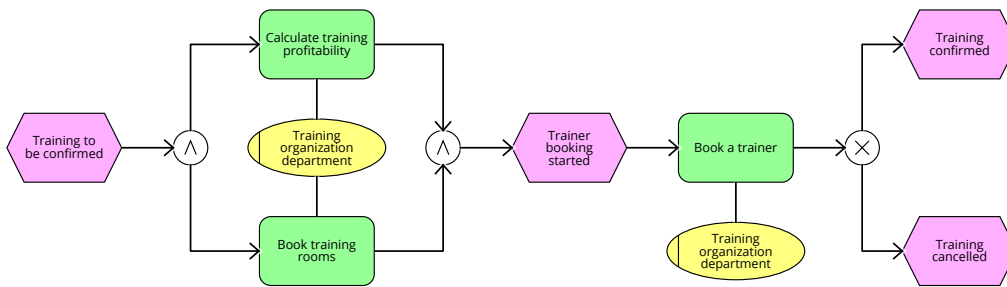

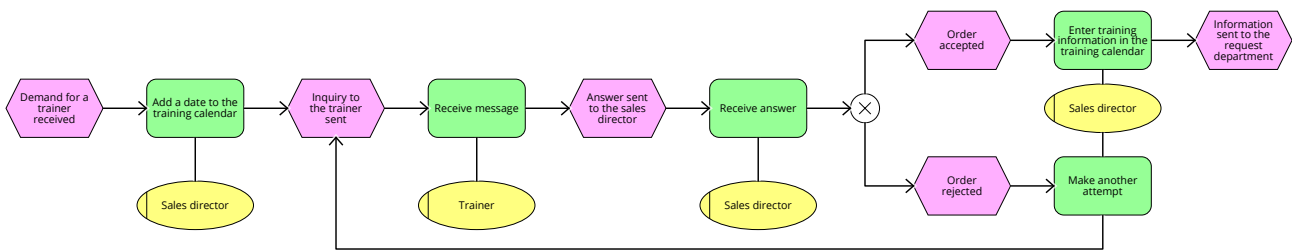Figure 8. Open training confirmation process EPC model.

Figure 9. Booking a trainer for training process EPC model.

The process of confirming open training usually starts seven days before the planned date. This is the moment when, based on specific factors, employees decide to implement the planned training, move participants to another date or cancel the training completely. The result of the whole process is a message directed to the client – *the date you signed up for has been confirmed, the training will take place* or *the date you have chosen has not been confirmed, the training has been canceled.*

The difference that should be mentioned at the beginning of the comparison is the color schemes. UML and BPMN do not use colored elements to prepare models, which are usually color-independent. However, according to EPC, it is good practice to use colors for specific groups of elements. All examples differ at the beginning: the start of the process

according to UML notation is initiated by the unlabelled point "start", and according to BPMN the initial event should be signed. Starting in a process prepared using EPC is presented using a named event. Sequence flow (in BPMN nomenclature) and control flows (according to UML and EPC) fall into operators that divide flows into two parallel paths. During their implementation, an attempt will be made to book a training room and the profitability of the training will be calculated. Income calculation and room booking are a series of separate activities, the process in question does not go into their details. BPMN notation, unlike the other two, offers a subprocess element (represented by a rectangle with a plus), which is used to present such cases [20]. In the first model, a fork divides the role of the operator dividing the flow, which is then merged. The BPMN diagram uses parallel gateways, and

the model prepared in EPC uses AND logical operators. In the first two models, flows continue to the next logical gateways. EPC notation, unlike BPMN and UML, does not allow logical operators to connect with each other, so the control flow leads into an event. At this point it is worth recalling that the EPC notation clearly defines what elements can be found at the output of logical operators. In the first and second examples, the places where the flow is split into two alternative paths, and the paths themselves are signed. The EPC did not accept labeling of logical operators or outgoing flows from them. In the BPMN scheme, one of the flows is marked with additional diagonal lines. This is the default process flow. UML and EPC notations do not have such a tool. The control flow in the UML model ends at a point called control end, the sequence flow in the BPMN model at the end event point. The EPC model began with an event and it must also end with one.

The second model represents the trainer booking process. It is an undertaking in which three organizational units of a training company are most often involved – customer service or sales department, sales director and a representative of the coaching team. These are separate departments, they work independently implementing their own processes. In this summary, it's worth looking at the ways in which notations it is possible to present roles in processes. The trainer's reservation is made in two cases - when the sales department representative finalizes the sale of the closed training and in cooperation with the client confirms the proposed dates or when the open training has implementation potential. This process always has one output – the selected lecturer accepts the order from the Commercial Director, the trainer is reserved, the selected date can be confirmed. The process begins with an e-mail request from a representative of one of the above departments. Thanks to BPMN notation, it is possible to clearly present the workflow or information between employees, individual organizational units of the enterprise or separate organizations. Aiming at a high level of detail, it was decided to apply them to this process. In the middle diagram, the responsibilities in the process are represented by pools. Each actor in the process – customer service, sales department, sales director or trainer – is presented as a separate pool with messages flowing between them. The UML activity diagram, unfortunately, does not offer such wide possibilities. In the EPC schema, determining the responsibility is possible using a graphic element organizational unit, while the notation for this element does not assume presenting message flows between units. The use of pools also enables choreography for the process. This is a separate type of process, which is representable only in BPMN notation.

VI. ANALYSIS OF THE SURVEY ORGANIZED AMONG THE STAFF OF THE TRAINING COMPANY

Eight employees participated in the study (about 30% of the company's full-time team). Respondents were presented with 6 models of business processes in UML, BPMN and EPC notation: open training confirmation and trainer booking. Study participants take part in these processes on a daily basis in the performance of their duties. Each interview lasted between 9

and 15 minutes, interviews were conducted individually with each participant. The respondent answered 9 questions related to the mentioned set of diagrams.

Only one respondent did not recognize the processes shown in the models. The others correctly named the diagrams presented to them and were able to embed processes in time – to determine the moment in the organization's activity when the process would be carried out.

The models were divided into two groups: one for each process. The employees were asked to choose the most readable diagram from each group that contained different diagram types. In the group of the open training confirmation process, the EPC model was most often selected, paying attention to its colors, which were to catch the eye. It was also claimed that thanks to the colors the model is more readable and helps to find in the process: "I look and know what's going on, I know where I am". After the detailed questions, it turned out that the subjects incorrectly interpret the meaning of colors in EPC schemes – one wrongly thought that one color was the task of one actor of the process or a given color means positive (confirmation of the training) and another negative (canceling the training). Two respondents chose the UML model praising the simplicity and transparency of performance, and only one respondent indicated the BPMN scheme. In the trainer reservation process group, four respondents decided to choose the BPMN model. Respondents praised the use of swimlanes, agreeing that with this more complex process involving several company departments, a clear presentation of responsibility plays an important role. Three respondents chose the EPC diagram, again paying attention to the attractive nature of the model's color scheme. It is worth mentioning that these respondents called themselves visual learners. Nobody decided to choose the UML activity diagram.

When asked about which model is the most understandable and useful for the respondents, for the first group, they most often chose the BPMN model. This diagram was indicated three times. The use of swimlanes and the variety of graphic elements used were appreciated. Two respondents chose UML models, paying attention to the exact and legible way of describing events. Others indicated EPC schemes. In the second group, six respondents considered the BPMN model to be the most understandable. Here again, a clear and accurate division of roles in the process was praised. One respondent chose the EPC scheme.

Respondents were asked to indicate the elements that they think make models less readable, drew attention to the use of colors in EPC notation. They were subjects who did not choose schemes prepared according to EPC notation, as well as people who were presented with the way in which they should interpret the colors of the graphic elements of this language. They also exchanged transverse lines placed on the flows presented in the BPMN models (default process flow), and also pointed out too extensive event names in UML. At the same time, the method of constructing messages in BPMN diagrams was appreciated.

In many cases, the respondents said that they did not

understand the meaning and, consequently, the actions of logical operators and gates and dashed lines in BPMN models (message flows). People familiar with logic understood the logical operators of EPC notation. Study participants complained about the lack of signatures under the elements 'start' and 'end' in UML diagrams, and completely non-intuitive 'fork' and 'control merging'. Everyone reported the need to present a legend by which they could learn the functions of unknown elements. When asked what could help them in understanding the process, they pointed to the need to sign logical operators in models for UML and BPMN, and to name the points initiating and terminating UML processes. In the answers to this question, the legend appeared again as an element necessary for the correct interpretation of processes.

Participants asked if they prefer models containing more or less details differed in the answers. The first respondent admitted that due to her professional experience and good knowledge of the presented processes, her basic model is sufficient. Others pointed out that models with a high level of detail do not leave room for their own interpretation, which is not recommended for self-organizing teams. They also claimed that by accurately presenting the process, it is easier to determine my place and responsibility, "I am able to be more independent and organize myself." Two respondents mentioned that models that included a large amount of details can be useful when introducing new employees to teams.

Respondents asked about whether they prefer colored (EPC diagrams) or black and white models mostly chose the former. However, they pointed out that elements that would be of the same color should mean the same or present the same process results (positive/negative result). They again admitted that a legend would be useful that would describe the meaning of individual colors. Two respondents said that colored schemes are easier to orientate and find, colors help categorize relevant groups of elements. Two respondents chose black and white process diagrams.

Respondents noted that models that clearly define responsibility for specific process steps are definitely more useful and make the schemes clearer. Everyone agreed that BPMN notation offering 'pool and track' tools handles role presentation best. The models in which the pools were used were assessed as the most clear, the participants immediately pointed out the clear division of roles in the process. The element of 'organizational units' was less often interpreted in an appropriate manner. The respondents did not immediately understand the function they play in the models.

VII. NOTATION INTEROPERABILITY

As there are several notations for business process modeling, the possibility of converting models between them is an important research topic.

In the case of transformations between BPMN and EPC, it is important to notice that one-to-one translation pattern cannot be used here, as there are syntax differences, e.g. in EPC each function must be followed by an event [32]. There are methods based on transforming rules from EPC to BPMN models [33].

However, during transformation from EPC to BPMN, the information content may change what may result in a slight information loss due to process model transformation [34]. Thus, recently new methods have been developed and new transformation rules proposed which should minimize the information loss [35].

There are also a number of papers concerning UML to BPMN [36]–[40] and BPMN to UML [41]–[43] transformation.

The conversions between the selected three notations can be dome through some other standard such as BPEL [44] or spreadsheets [45]. Workflow patterns constitute another effective way of transformation. However, as noticed by Grigorova and Mironov [46], [47], there is still a gap between some pattern constructs available in the notations, such as cancel activity (not supported by EPC), persistent trigger and generalized AND (not supported by UML).

Khudori and Kurniawan [48] conducted a broad survey on business process transformation techniques and stated that none of the existing techniques supports a truly complete transformation between the process modeling notations.

VIII. CONCLUSIONS

In this work, we discussed and compared the three most popular business process modeling notations — Unified Modeling Language, Business Process Model and Notation and Event-Driven Process Chain. Our work included also a comparative study of the above-mentioned notations and a survey conducted among the employees of a mid-sized training company, as well as an overview of notation interoperability approaches present in the current research work.

Unified Modeling Language can be successfully used to prepare business process models. It offers two diagrams that can be used for this purpose: use cases and activities. The latter presents the process as a sequence of actions and actions that the control flow connects to decision blocks.

Business Process Model and Notation is the richest notation in terms of the number of graphic elements. Thanks to this, the person preparing models can very accurately reflect the nature of the designed activity or event, present the type of task being performed, mark the default process flow or present the required data objects. BPMN offers also a way to present roles in the process in a very clear and legible way to the user, in form of pools and lanes. It also lets users create choreography diagrams that present information flows and interactions between pools in a more user-friendly way.

Event-Driven Process Chain is characterized by a relatively small number of elements that can be used to build a diagram, as well as the practice of using colors. After all, it allows users to create models even for complex business processes, while assigning responsibilities to specific functions, using organizational units.

As future work, we plan to calculate and compare complexity metrics of the analyzed business process models, as well as to conduct a more detailed survey among a larger set of companies from the SME sector.

REFERENCES

[1] P. Śliż, "Robotization of business processes and the future of the labor market in poland-preliminary research," *Organization and Management*, no. 2 (185), pp. 67–79, 2019.

[2] R. Gabryelczyk, *ARIS w modelowaniu procesów biznesu*. Difin, 2006.

[3] OMG, "Unified Modeling Language (OMG UML) version 2.2. super-structure," Object Management Group, Tech. Rep. formal/2009-02-02, February 2009.

[4] J. Hunt, *Guide to the Unified Process featuring UML, Java and Design Patterns*. Springer, 2003.

[5] M. Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3rd ed. Addison-Wesley Professional, 2003.

[6] D. Pilone and N. Pitman, *UML 2.0 in a Nutshell*. O'Reilly, 2005.

[7] M. Owen and J. Raj, "BPMN and Business Process Management. Introduction to the new business process modeling standard." OMG, Tech. Rep., 2006, www.bpmn.org.

[8] N. Russell, W. M. P. van der Aalst, A. H. M. ter Hofstede, and P. Wohed, "On the suitability of UML 2.0 activity diagrams for business process modelling," in *Proceedings of the 3rd Asia-Pacific conference on Conceptual modelling – Volume 53*, ser. APCCM '06. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2006, pp. 95–104.

[9] D. Jäger, A. Schleicher, and B. Westfechtel, "Using uml for software process modeling," in *Software Engineering – ESEC/FSE '99*, ser. Lecture Notes in Computer Science, O. Nierstrasz and M. Lemoine, Eds. Springer Berlin Heidelberg, 1999, vol. 1687, pp. 91–108.

[10] G. Engels, A. Förster, R. Heckel, and S. Thöne, "Process modeling using UML," *Process-Aware Information Systems*, pp. 85–117, 2005.

[11] M. Razavian and R. Khosravi, "Modeling variability in business process models using UML," in *Proceedings of the fifth International Conference on Information Technology: New Generations, 2008. ITNG 2008*, 2008, pp. 82–87.

[12] M. A. Kose and M. Ozkaya, "Towards extending uml's activity diagram for the architectural modeling, analysis, and implementation," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2020, pp. 639–648.

[13] A. Derezińska and Ł. Zaremba, "Approaches to semantic mutation of behavioral state machines in model-driven software development," in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2018, pp. 863–866.

[14] F. Hunka and J. Matula, "Towards paired transactions modeling," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2016, pp. 1153–1158.

[15] T. Gzik, "Modelowanie procesów biznesowych w UML," *Business Process Management Portal*, 2018.

[16] K. Kluza, P. Wiśniewski, K. Jobczyk, A. Ligęza, and A. Suchenia (Mroczek), "Comparison of selected modeling notations for process, decision and system modeling," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*. IEEE, 2017, pp. 1095–1098.

[17] M. Chinosi and A. Trombetta, "BPMN: An introduction to the standard," *Computer Standards & Interfaces*, vol. 34, no. 1, pp. 124–134, 2012.

[18] P. Y. H. Wong and J. Gibbons, "Formalisations and applications of bpmn," *Science of Computer Programming*, vol. 76, no. 8, pp. 633–650, 2011.

[19] OMG, "Business Process Model and Notation (BPMN): Version 2.0 specification," Object Management Group, Tech. Rep. formal/2011-01-03, January 2011.

[20] S. Drejewicz, *Zrozumieć BPMN modelowanie procesów biznesowych*. Wydawnictwo Helion, 2012.

[21] *Business Process Model and Notation (BPMN)*, 2nd ed., Object Management Group, 12 2013, version 2.0.2 contains a minor change to Clause 15.

[22] E. M. Sanfilippo, S. Borgo, and C. Masolo, "Events and activities: Is there an ontology behind BPMN?" in *FOIS*, 2014, pp. 147–156.

[23] A. W. Scheer, *Aris: Business Process Modeling*, 3rd ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2000.

[24] M. Rosemann and W. M. P. van der Aalst, "A configurable reference modelling language," *Information Systems*, vol. 32, no. 1, pp. 1–23, 2007.

[25] A. Amjad, F. Azam, M. W. Anwar, W. H. Butt, and M. Rashid, "Event-driven process chain for modeling and verification of business requirements–a systematic literature review," *IEEE Access*, vol. 6, pp. 9027–9048, 2018.

[26] D. M. Riehle, S. Jannaber, A. Karhof, O. Thomas, P. Delfmann, and J. Becker, "On the de-facto standard of event-driven process chains: How epc is defined in literature," *Modellierung 2016*, 2016.

[27] W. M. P. van der Aalst, "Formalization and verification of event-driven process chains," *Information and Software Technology*, vol. 41, no. 10, pp. 639–650, 1999.

[28] W. M. P. van der Aalst, J. Desel, and E. Kindler, "On the semantics of EPCs: A vicious circle," in *Proceedings of the EPK 2002: Business Process Management Using EPCs*, Bonn, Trier, Germany, November 2002, pp. 71–80.

[29] P. Pasamonik, "Modelowanie procesów biznesowych zorientowane na czynności," *Zeszyty Naukowe Wyższej Szkoły Informatyki*, vol. 9, no. 2, pp. 102–116, 2010.

[30] A.-W. Scheer, *Architecture of integrated information systems: foundations of enterprise modelling*. Springer Science & Business Media, 2012.

[31] P. B. Kruchten, "The 4+ 1 view model of architecture," *IEEE software*, vol. 12, no. 6, pp. 42–50, 1995.

[32] M. Cheung and J. Hidders, "Round-trip iterative business process modelling between bpa and bpms tools," *Business Process Management Journal*, 2011.

[33] W. Tscheschner, "Transformation from epc to bpmn," *Business Process Technology*, vol. 1, no. 3, pp. 7–21, 2006.

[34] O. Levina, "Assessing information loss in epc to bpmn business process model transformation," in *2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops*. IEEE, 2012, pp. 51–55.

[35] A. N. Khudori and T. A. Kurniawan, "Transforming epc aris markup language into bpmn metadata," in *2019 International Conference on Sustainable Information Engineering and Technology (SIET)*. IEEE, 2019, pp. 358–363.

[36] J. Chanda, A. Kanjilal, S. Sengupta, and S. Bhattacharya, "Fam2bp: Transformation framework of uml behavioral elements into bpmn design element," in *International Conference on Computer Science and Information Technology*. Springer, 2011, pp. 70–79.

[37] Y. Wautelet and S. Poelmans, "An integrated enterprise modeling framework using the rup/uml business use-case model and bpmn," in *IFIP Working Conference on The Practice of Enterprise Modeling*. Springer, 2017, pp. 299–315.

[38] A. Kalnins and V. Vitolins, "Use of UML and model transformations for workflow process definitions," *arXiv preprint cs/0607044*, 2006.

[39] M. Argañaraz, A. Funes, and A. Dasso, "An mda approach to business process model transformations," *Electronic Journal of SADIO (EJS)*, vol. 9, pp. 24–48, 2010.

[40] J. Pulgar and M. C. Bastarrica, "Transforming multi-role activities in software processes into business processes," in *International Conference on Business Process Management*. Springer, 2016, pp. 372–383.

[41] M. A. Cibran, "Translating BPMN models into UML activities," in *International Conference on Business Process Management*. Springer, 2008, pp. 236–247.

[42] N. Q. Bao, "A proposal for a method to translate BPMN model into UML activity diagram," in *13th International Conference on Business Information Systems*, 2010.

[43] L. Aversano, C. Grasso, and M. Tortorella, "Managing the alignment between business processes and software systems," *Information and Software Technology*, vol. 72, pp. 171–188, 2016.

[44] K. Grolinger, M. A. Capretz, A. Cunha, and S. Tazi, "Integration of business process modeling and web services: a survey," *Service Oriented Computing and Applications*, vol. 8, no. 2, pp. 105–128, 2014.

[45] P. Wiśniewski, K. Kluza, E. Kucharska, and A. Ligęza, "Spreadsheets as interoperability solution for business process representation," *Applied Sciences*, vol. 9, no. 2, p. 345, 2019.

[46] K. Grigorova and K. Mironov, "Bridging the gap between different interfaces for business process modeling," *International Journal of Computer and Information Engineering*, vol. 9, no. 12, pp. 2479–2482, 2015.

[47] ——, "Conversion of business process models using workflow patterns," in *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2018, pp. 763–766.

[48] A. N. Khudori and T. A. Kurniawan, "Business process model transformation techniques: A comprehensive survey," *Advanced Science Letters*, vol. 24, no. 11, pp. 8606–8612, 2018.