Designing executable business processes as a programming paradigm

Andrey G. Mikheev
Associate Professor, Department of Business Informatics and Industrial Management Systems
National University of Science and Technology MISiS
Address: 4, Leninsky Prospect, Moscow, 119991, Russian Federation
E-mail: andrmikheev@gmail.com

Valeriy E. Pyatetskiy
Professor, Head of Department of Business Informatics and Industrial Management Systems
National University of Science and Technology MISiS
Address: 4, Leninsky Prospect, Moscow, 119991, Russian Federation
E-mail: 7621496@gmail.com

Abstract

This article discusses techniques used to design business processes that are directly executable on the computer system of an enterprise (executable business processes). It also describes the experience of teaching the elements of this technology. This experience was accumulated within two years of teaching process disciplines to bachelors and masters in National University of Science and Technology MISiS and Moscow State University of Economics, Statistics and Informatics (MESI).

One of the reasons to choose the process way of enterprise automation is reducing the cost of automation. In traditional automation, at first the business analyst describes the functionality of the designed system in the form of text, then the programmer translates it into code. The use of executable business processes would make it possible to avoid duplication of work in many ways. In this case the business analyst with the customer uses visual graphic software to develop the business processes of automated functionality which will then be executed directly in the computer environment. Schemes of executable business processes are the human-readable graphical description of the corresponding functionality and it's not necessary to translate them into code. Therefore, the cost of analytical work in this case is the same while the cost of programming is significantly lower. If the business environment changes, the business analyst can quickly change the schemes of business processes accordingly without involving the programmer. In addition, in many cases, the business analyst can independently (without programmer) develop new business processes. Therefore, the cost of development, maintenance and support of such IT-solutions is significantly lower than the cost of traditional solutions, while the speed of development, implementation and subsequent changes is significantly higher.

These advantages (faster, cheaper, easier to maintain and support) are the same advantages the paradigm of object-oriented programming has over the procedural programming paradigm. By analogy, we can call the development of software solutions based on executable business processes a new programming paradigm with respect to the traditional approach.

Process automation based on executable business processes requires process thinking from business analysts that differs from the thinking of IT specialists in the traditional enterprise automation. In addition to knowledge of business process notations, business analysts should be able to implement the typical situations in enterprise business in the form of executable business processes. This article presents the methodology that was used to teach students the process thinking.

Introduction

Notwithstanding the fact that the first analogues of modern business process management systems (at that time they were called workflow-systems [1]) appeared more than fifteen years ago, until recently, most of the process management works (for example, [2 − 4]) were limited by the study of production activity of enterprises, identification of repetitive chains of actions, formalization and integration of these chains into completed business processes, analysis of selected business processes, and development of recommendations for changes in business processes so that at the same time the operating efficiency of the enterprise increased. This did not imply automated execution of business processes. The use of computer systems was limited only to business process modeling. That is, after the development or modification of a business process it was introduced to organizations by administrative methods, which are long, clumsy and expensive.

In recent years, quality changes occurred in this field [5]. Currently, enterprises have been actively implementing computer systems directly executing business processes in the computer environment which are called the business process management system (BPMS). These systems distribute tasks to the executors and monitor their implementation. The sequence of tasks is determined by the business process diagram. Control points move across the diagram; in the design nodes control points generate tasks for the executors.

Thus, at an "office"-type enterprise, an analogue of the production line appears: this mechanism makes it possible to exclude routine operations from the employees' actions, inefficient procedures related to information search and transmission, and significantly to increase the rate of employee interaction. This is due to the fact that by using BPMS, employees accomplish received tasks sticking to receiving data required for task execution from other employees; they transmit the results of their work to other employees; and they study the job descriptions. All information needed to perform the task appears on the employee's computer screen.

At enterprises with stable recurring chains of operations, the use of BPMS provides other advantages:

✦ significantly simplifies the control activities for works in progress and increases the transparency of business operations;

✦ improves the enterprise product quality: through automatic regulatory activity and monitoring tools to observe all rules provided

✦ makes it possible to promptly change business processes in response to the changing business environment of the enterprise;

✦ makes it possible to solve the problem of enterprise-scale integration;

✦ reduces the cost of enterprise automation, and improves the rate of software development and reliability.

Let us explain the last item in the list. Reducing costs is one of the reasons for selecting a process automation option. In the traditional approach, at the beginning a detailed technical project (in the form of a plain text) is drawn up. This is approved by the customer, and subsequently it is converted to program code by software programmers. Automation based executable business processes makes it possible to eliminate duplication: in this case, the analyst immediately develops executable business processes, which are approved by the customer and do not have to be translated into a program code. Therefore, the development time and costs of the work of the executors are significantly reduced.

Automation based on executable business processes makes it possible to quickly adapt the development to changing problems and new ideas started up in the course of development, to reduce development costs, to reduce the technical support costs, and significantly to reduce the cost of modifications and maintenance.

Thus, system implementation, customization and management based on executable business processes prove to be faster and cheaper as compared to traditional automation in which separate application components are developed for various problems and functions. These advantages (faster, cheaper, easier to support and maintain) coincide with the paradigm advantages of object-oriented programming as compared to procedural programming. By analogy, the activity of designing executable business processes can be called a new programming paradigm.

In this case, the concept of the paradigm is considered in terms of R.Floyd's programming paradigm concept [6], which is an extension of T.Kuhn's paradigm concept proposed in the paper [7].

In recent years, executable business processes have been actively implemented both in business and government organizations. However, automation based on executable business processes requires process thinking of specialists different from thinking of IT specialists using the traditional enterprise automation. Apart from the knowledge of notations describing business processes and interfaces, BPMS, business analysts should be able to implement some business specific situations in the form of executable business processes. In this regard, a task appears relating to training specialists for both eco-

nomic and information technology-related specialties in the process approach and BPMS handling.

This article offers rules of outlining business processes, and it provides solutions for some typical situations. These rules can be considered as an extension and refinement of a set of rules set forth in the section 'BPMN Best Practices' [8].

1. Training for executable
business processes development

The paradigm of object-oriented programming has led to the emergence of specialists whose way of thinking deeply varies from the traditional thinking of procedure-oriented software programmers. Making comparisons with process automation, it is fair to say that after progressing to a certain stage of the business, rapidly growing business processes-based automation will require a large number of specialists, i.e. business analysts possessing process thinking.

Even today these specialists have to be trained in higher education institutions. By analogy with programming training, teaching students business processes development can be divided into two approaches:

❖ study of business process description notations and training to work with specific BPMS (similar to learning the syntax of programming languages and specific compilers)

❖ study of various possible implementation options in the form of executable business processes of various typical situations in the enterprise business (similar to learning programming techniques)

There is a large number of training courses dedicated to the first approach. For example, papers [9-10] summarize the experience of teaching students to develop executable business processes in MESI, NITU MISiS and UGATU. In the lessons students learn the theory of executable business processes, graphical business process description notations (the most popular notation is BPMN, but sometimes UML AD is used [11]), the main components of typical BPMS, and they acquire practical experience in development and execution of the simplest business processes.

In the course of training, the issues of handling transient business processes, rules of selecting traffic of control points and capabilities of assigning terms of job execution are studied and consolidated in practice. The developed business processes are executed by students under different roles in the software environment.

Training courses dedicated to construction of various process automation solutions based on executable business processes are currently still being established. Within the framework of such courses, on the basis of agreed sets of specially selected business situations and options of process solutions for them, techniques of outlining effective solutions can be taught, and the process thinking of students can be developed. The following section provides examples of situations, and it formulates rules of constructing business process diagrams developed in BPMS operating practice at enterprises.

2. Engineering practice
of executable business processes

2.1. Formulations used in the names of business process diagram nodes appropriate to executor actions

The name of a node, which carries a job for the executor, in a majority of BPMS is identical to the name of the job that is displayed to the user. The jobs should be formulated so that they are clear to the extent possible to the executor. From the authors' experience, the clearest are wordings including an infinitive and a noun, such as "issue an order", "review the application". In the lessons conducted by the authors, this kind of naming business process nodes is mandatory.

2.2. Size of business process diagram

It is extremely difficult to analyze business process diagrams having a size larger than one and a half times the size of the computer screen. If the diagram does not fit on the screen, it is necessary to try to move its parts into internal or external sub-processes.

2.3. Motion direction of control points across the business process diagram

It is comfortable to analyze motion of control points across the business process diagram when a general motion of control points corresponds to motion of the area which a person looks at while reading texts. Therefore, it is desirable to place the business process diagram nodes so that the general motion of the control points in them go from left to right or from top to bottom. In case of long sections of control point motion, the diagram nodes connected by junction lines can be arranged from left to right and from top to bottom much as a person reads words on a sheet of paper document (*Figure 1*).

In case of complicated behavior logic of a business process, when a large number of loops of directed transitions appear in the diagram, this cannot be achieved. However, the majority of business processes used in practice have a simple logic, and in their development we need to pay attention to correspondence of the overall motion of control points in the selected direction (from left to right or from top to bottom).
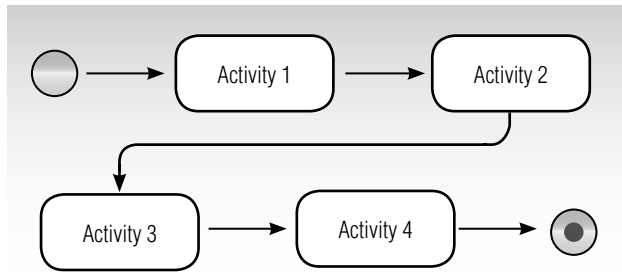
Fig. 1. Diagram of control point motion from left to right
and from top to bottom

2.4. Do not use roles in the form of swimlines in business process diagrams

The roles are intended to link the business process diagram nodes with the task executors. The majority of modern graphical notations makes it possible to assign roles in the diagram in the form of horizontal or vertical stripes, called swimlines. In this case, all diagram nodes located in the swimline are associated with a role corresponding to the track.

The practical experience of the authors shows that the use of roles in the form of swimlines is inconvenient in the industrial business processes of the enterprise, inasmuch as the need for placing the business process diagram nodes on a certain strip prevents your developing diagrams that are easy-to-understand in terms of motion of the control points, and it also significantly increases the area occupied by the business process diagram.

Information on a node-related role is important to analyze the business process diagram. Therefore, it is proposed to put the role name in parentheses at the top of the graphic element of the action node and consider it as a prefix of the node name. This technique will be used in the diagrams provided in this article.

2.5. Implementation of actions to be simultaneously performed by two executors

In some cases, the action should be simultaneously performed by two executors (for example, one employee should sign a document which is in the possession of the other employee). As a rule, the intuitive realization of such a scenario corresponds to a sequential arrangement of two nodes in the business process diagram; in so doing, the executor in the first node is an employee who should sign the document, and the second executor is an employee who is in possession the document is (*Figure 2*).
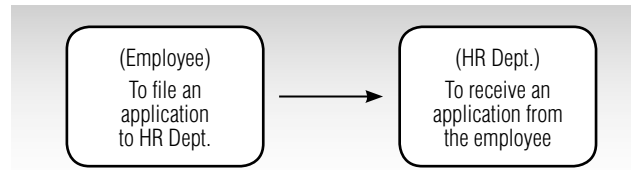

Fig. 2. "Intuitive" implementation of an action performed
simultaneously by two persons

However, the practice of business process maintenance shows that such a solution fails, since such node arrangement does not allow for coordinating the interaction. In this case, it is proposed that the nodes be placed in parallel (*Figure 3*).
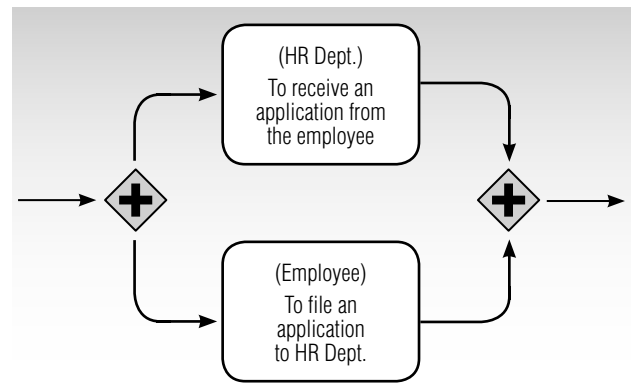

Fig. 3. Proper implementation of an action
being performed simultaneously by two persons

2.6. Taking secondary actions to a parallel branch

Let us consider the case when several consecutive actions should be performed simultaneously by two executors. The practice of executable business processes shows that the roles of officials, such as the "accountant" or "cashier" correspond to "responsible" employees, and the role of "employee" or "applicant" corresponds to much lesser "responsible" employees who can forget to mark the job processing for weeks.

Figure 4 provides an example of a business process diagram in which the tasks, for which execution is introduced by an "employee" into BPMS, can disable the normal business process flow. These tasks are marked in the Figure by oval curves. For example, if an employee does not mark a job execution in BPMS "become familiar with the approval", the business process will not proceed to execution of the order and disbursement of money. In the commercial operation mode, such diagrams of business processes can lead to serious disruptions in enterprise performance.
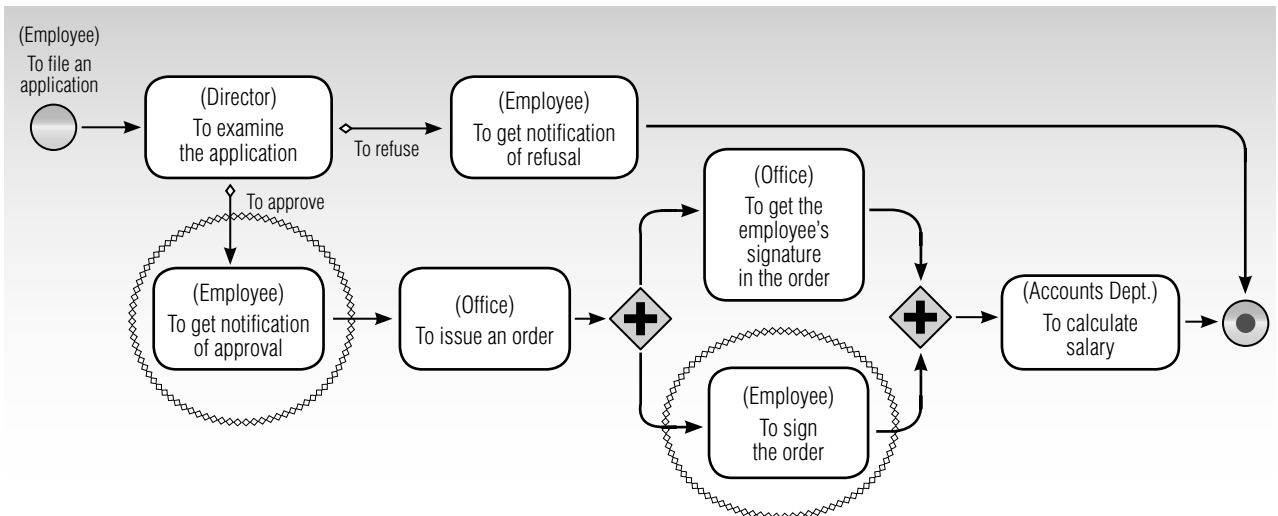
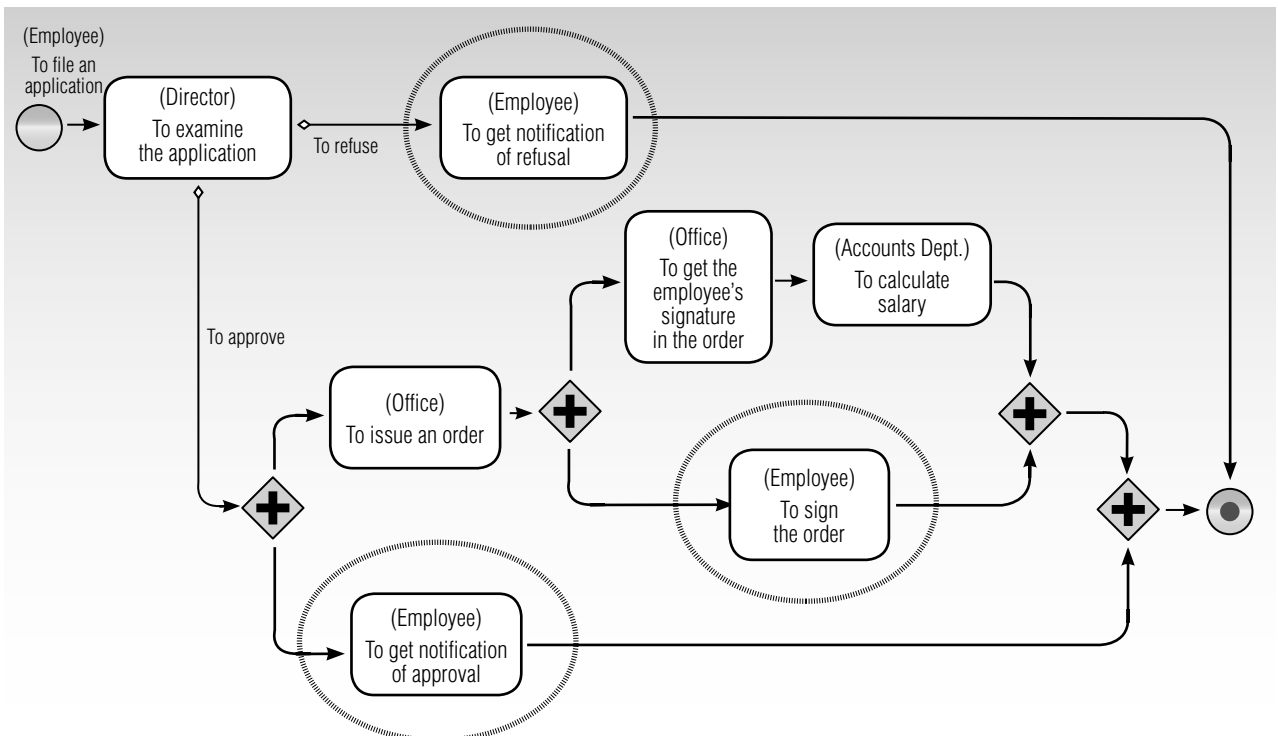*Fig. 4.* Improper implementation of the business process with secondary actions

Fig. 5. Correct implementation of a business process with secondary actions

Therefore, the business process diagram is to be drawn out so that the secondary jobs executed by the employee do not suspend further implementation of the business process. Each such job should be performed in a parallel branch, and after it no essential tasks of the business process should be performed. An example of a correct outlining of the business process diagram corresponding to the diagram actions in *Figure 4* is provided in *Figure 5*.

2.7. Using paired splits and merges: Implementation of the possibility of decomposing a diagram section

In complicated cases, split elements without their paired elements — merges (split is what we call a parallel gateway with one incoming and many outgoing transitions, and merge is what we call a parallel gateway with multiple incoming and one outgoing transitions) are
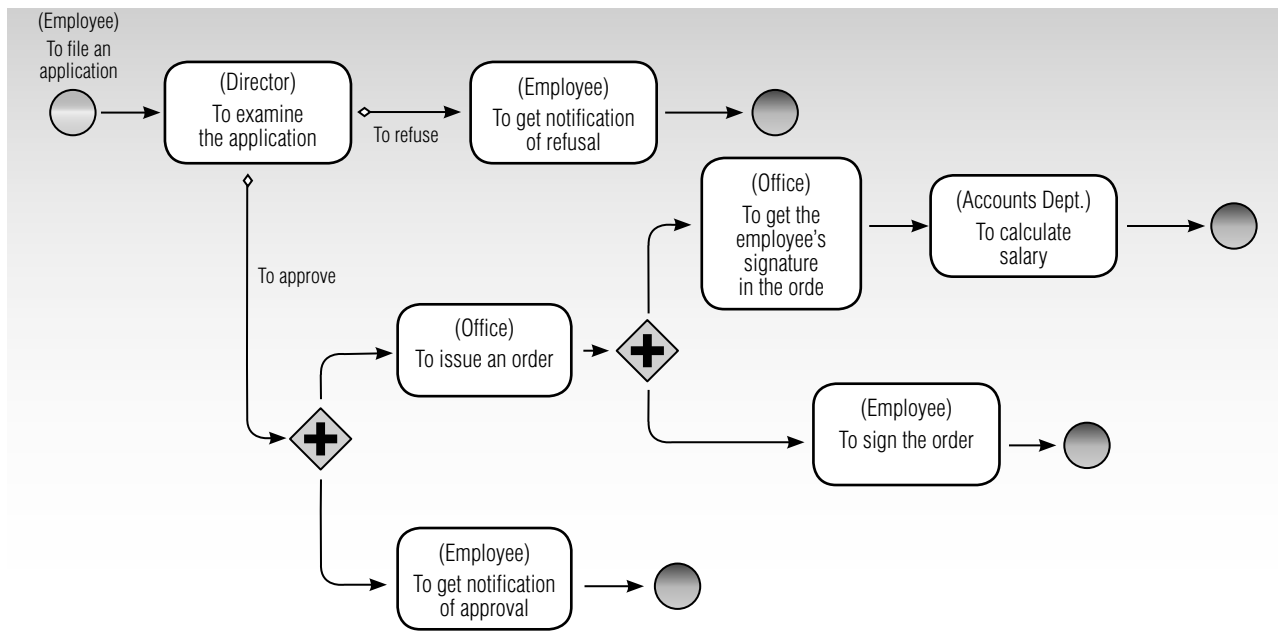
Fig. 6. Alternative implementation of a business process with secondary actions
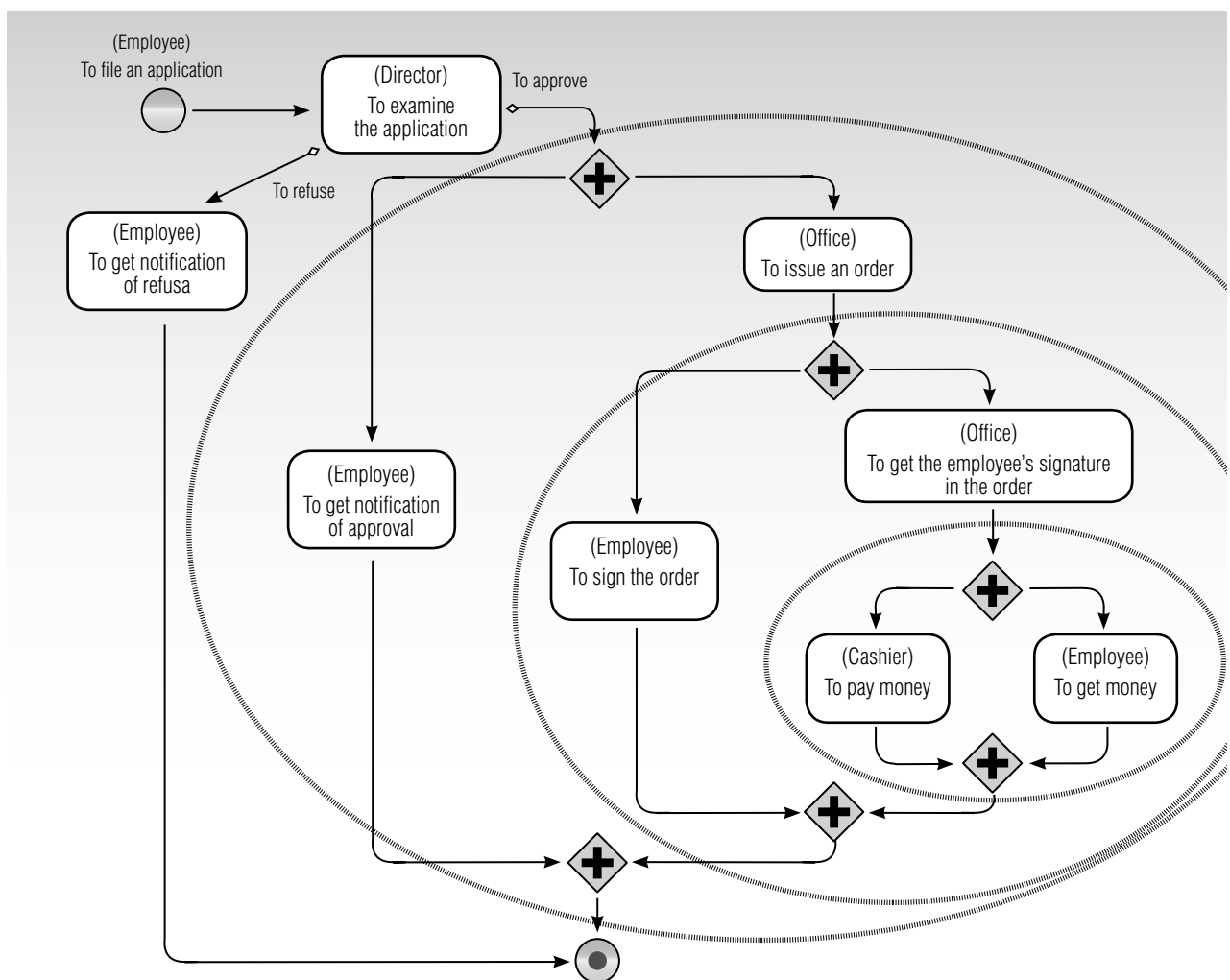

Fig. 7. Example of a business process diagram with three embedded split and merge pairs

sometimes used in the business process diagrams. An element "end of control flow" is used in such diagrams to remove the control points which completed their job (*Figure 6* shows an example of diagram with unpaired elements equivalent to the diagram in *Figure 5*). However, as noted in paper [8], a preferred diagram is a diagram with paired splits and merges, since such diagrams are easier to understand.

This happens because the diagram section between the split and its paired merge can be mentally decomposed and, thus, the business process diagram is split into two simpler diagrams. Having experience, the business analyst can quickly "read" such diagrams. In case of large diagrams with unpaired elements, the business analyst has to "decode" these diagrams; that requires much more time and effort. *Figure 7* shows a diagram with three embedded split and merge pairs. It is evident that the diagram drawn out in this way enables us to mentally decompose it successively three times and, thus, to simplify the complexity of its visual perception.

2.8. Location of paired splits — merges and connecting transitions

It is convenient to locate splits and their paired merges on the same horizontal or vertical line so that a paired element for one element can be easily found in the business process diagram. It is desirable that the transition lines corresponding to simultaneously running action flows be parallel. This makes it easier to understand the diagram, as it is easier for the business analyst to arrange

sequences of actions in the diagram in parallel as "running in parallel." Examples of such arrangements are shown in *Figures 5* and *7*.

2.9. Use of the "end of business process" element

It is preferable to use "end of the business process" elements rather than "end of control flow" elements, because in this case, a business analyst can more easily analyze the chart of a workflow instance process being performed with control points posted thereon. Once a control point arrives at an "end of business process" element, the workflow instance is immediately completed, whereas when an "end of control flow" element is used, a business analyst has to expend more effort to ensure that all control points have come to "end of control flow" elements.

Based on the "end of the business process" element, one can build process solutions for certain situations. Let us consider the case of document concurrence: three departments should agree on the document. Each department may approve or reject the document. If any of the departments rejects the document, the document receives the status "not agreed" and the concurrence should stop immediately, since there is no need for other departments to review the documents.

Concurrence by all departments should be done in parallel. The approval procedure in the business process is not important. *Figure 8* provides a diagram which when used within the sub-process solves the task set.
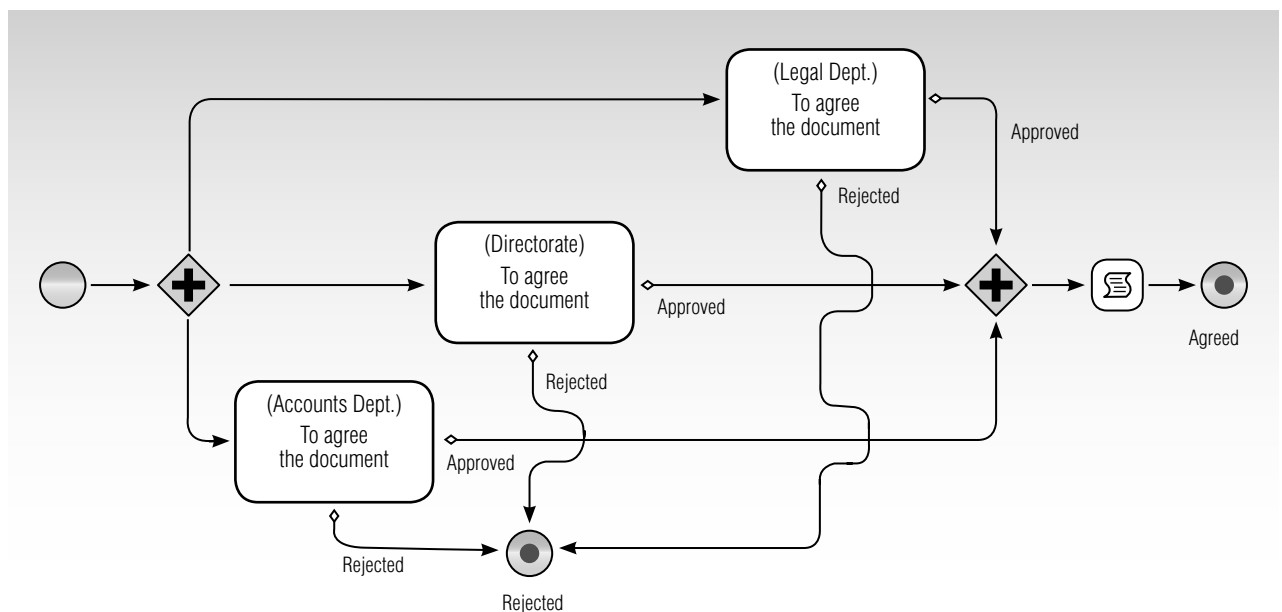

Fig. 8. The business process diagram implementing document concurrence

According to this diagram, any control point which comes to an end-node "Rejected" due to a rejection by any department will stop implementation of the entire sub-process and, in particular, remove the control points from the nodes in which document was concurred with two other departments. In case of a positive decision made by any department, the control point gets into a merge element, in which "it is waiting" for positive solutions from other departments.

2.10. An example of a compromise solution on splits — merges and use of the "end of control flow" element

The development of executable business processes is an art like an art of traditional programming. There are no ready recipes for all possible situations. In many cases, the solution proves to be compromise schemes combining both recommendations proposed in this article, and some exceptions.

As an example, *Figure 9* depicts a simplified business process diagram of a retail lending bank. It contains both paired splits and merges or unpaired splits, and an end node of business process, and two end nodes of the control flow. It goes like this, because if the credit manager

rejects it, there will be another executer who will also have to be informed about the rejection.

2.11. Use of algorithms in the business process diagram

Due to the fact that business process diagrams are very similar to control flow charts, a solution algorithm for a certain problem can be included directly in the business process diagram. This approach can be applied in development of both industrial and educational business processes.

As an example, let us consider a process implementation of the classical M.Gardner's problem of a discerning bride as formulated by academic E.B.Dynkin [12]. Grooms one by one come to one bride; the total number of grooms is known in advance. She keeps company with each of them no more than once and can compare a groom with any of the previous ones. If she selects someone, the selection process is terminated. If the bride rejects someone, she cannot meet him again. The bride's intention is to select the best groom with a maximum probability.

Figures 10 and *11* provide a business process and its sub-processes implementing this task.
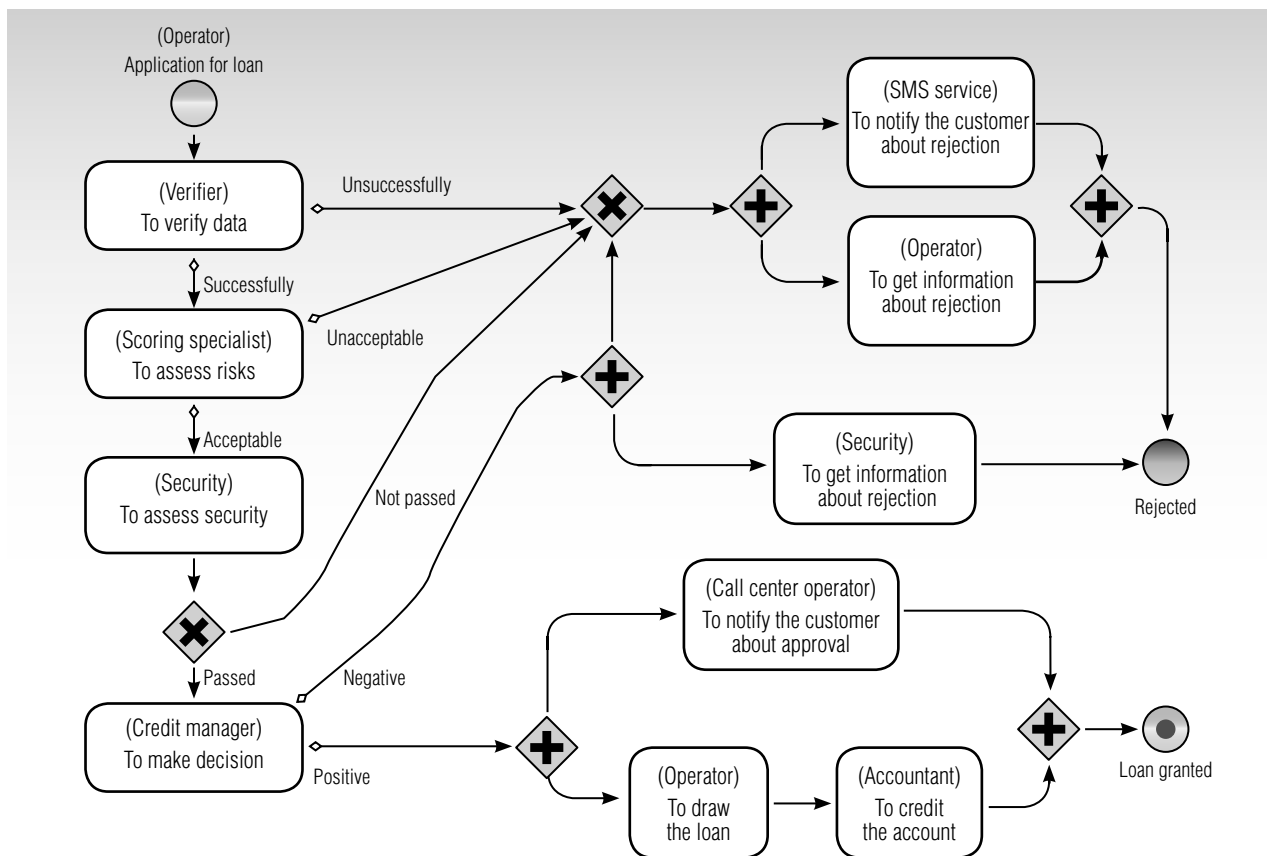

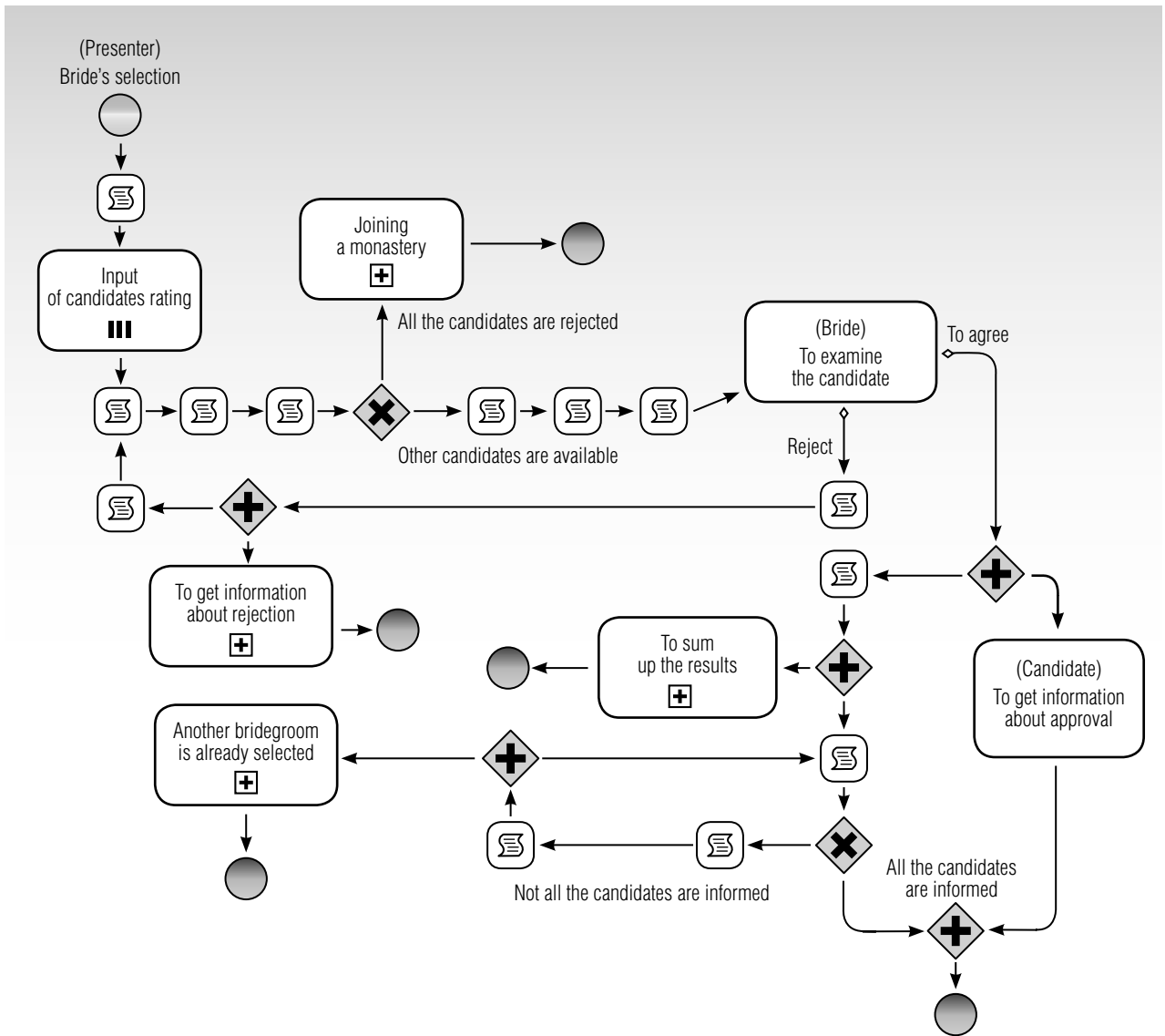*Fig. 9.* Business process diagram of bank retail lending

Fig. 10. Business process diagram of the discerning bride problem

In the business process diagram, tasks-scripts (small-size elements) correspond to support operations such as determining the number of marriage candidates in the list of candidates, extracting the current marriage candidate from the list of candidates, replacing the current marriage candidate with the next candidate in the list, etc.

3. Application of free software with an open code to train specialists in process automation

To teach students process automation, courses [9, 10] apply free software – BPMS RunaWFE [13]. The application of the free software for education makes it easy to introduce a training course in the educational process of any Russian university: it is free, available online at the project website RunaWFE (*http://runawfe.org/rus*). No keys or license files are required to install the system. The number of installations is not limited.

Conclusion

This article describes business situations for which solutions are proposed in the form of executable business processes. In addition we offer rules of constructing business process diagrams and an approach for training specialists – business analysts. The cited examples show that the development of executable business processes is a new area of activity for which new methods of building IT solutions and new training procedures should be created. ■
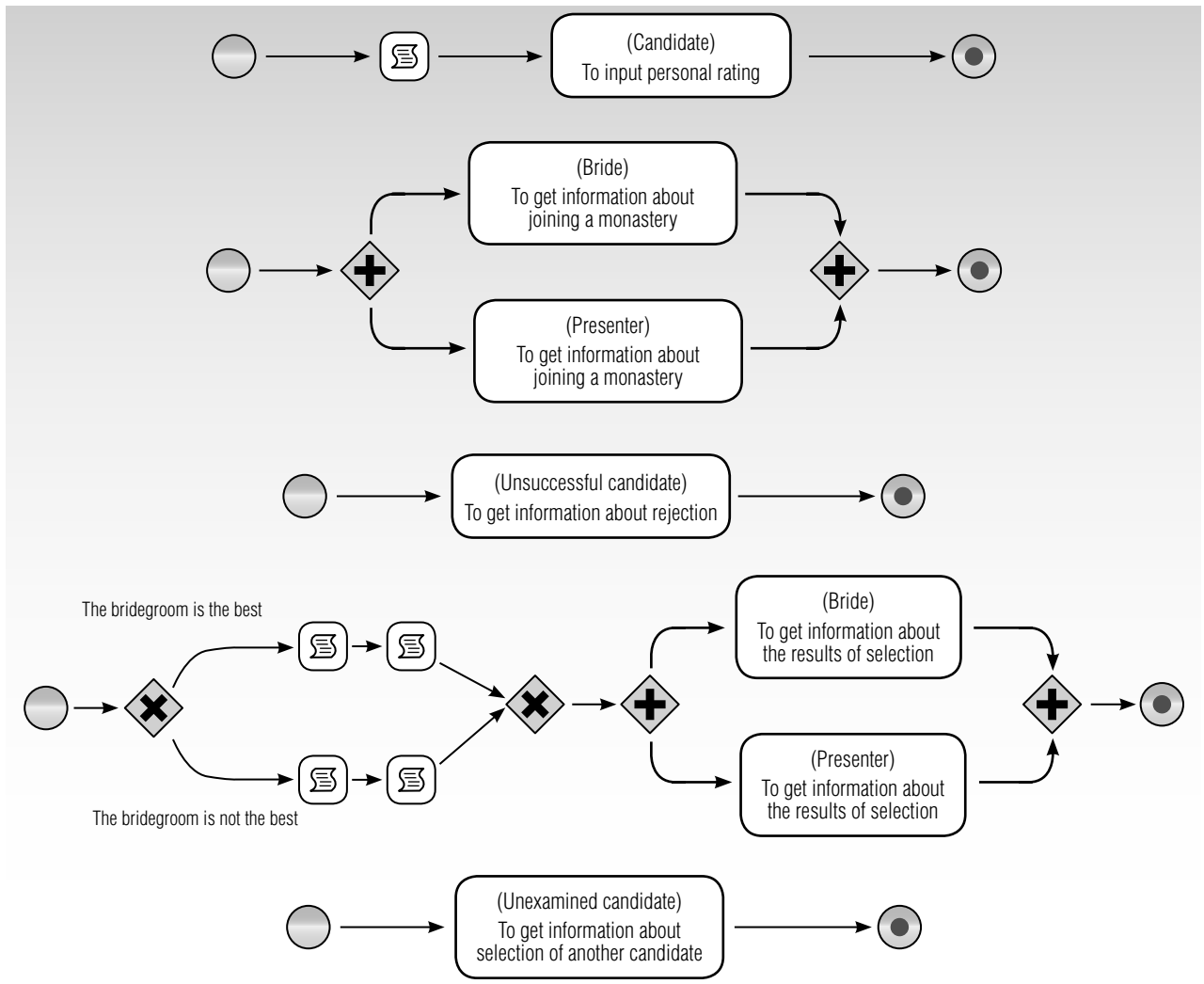
Fig. 11. Sub-process diagrams of the business process of the discerning bride problem

References

1. Van der Aalst W., Van Hee K. (2004) *Workflow management: Models, methods and systems.* Cambridge, MA: MIT Press, 2004.
2. Abdikeev N.M., Danko T.P., Ildemenov S.V., Kiselev A.D. (2005) *Reinzhiniring biznes-protsessov* [Business process reengineering]. Moscow: Eksmo (in Russian).
3. Telnov Yu.F. (2004) *Reinzhiniring biznes-protsessov. Komponentnaya metodologiya* [Business process reengineering. Component methodology]. Moscow: Finance and Statistics (in Russian).
4. Kalyanov G.N. (2006) *Modelirovanie, analiz, reorganizatsiya i avtomatizatsiya biznes-protsessov* [Business process modeling, analysis, reorganization and automation]. Moscow: Finance and Statistics (in Russian).
5. Vagner Yu.B. (2009) BPMS-effect [Effect ot BPMS]. *Automation in Industry*, no. 7, pp. 42−47 (in Russian).
6. Floyd R.W. (1979) The paradigms of programming. *Communications of the ACM*, no. 22 (8), pp. 455−460.
7. Kuhn T.S. (1962) *The structure of scientific revolutions.* Chicago: University of Chicago Press.
8. White S.A., Miers D. (2008) *BPMN modeling and reference guide: Understanding and using BPMN.* Lighthouse Point, FL: Future Strategies, 2008.
9. Kulikov G.G., Mikheev A.G. (2011) Osobennosti realizatsii protsessnogo podhoda I obuchenija upravleniu biznes-protsessami pri pomoshi svobodnogo PO s otkrytym kodom [Particularities of implementation of process approach and business processes management using free software]. *Open Education*, no. 4, pp. 47−57 (in Russian).
10. Pyatetskiy V.E., Mikheev A.G., Novichihin V.V (2013) *Sistema upravlenija biznes-protsessami: osnovy razrabotki biznes-protsessov pri pomoshi svobodnogo programmnogo obespechenija* [Business process management system: The basics of business process design using free software]. Moscow: MISiS (in Russian).
11. Fiodorov I.G. (2011) Sravnitelnyj analiz notatsij modelirovanija biznes-protsesov [Comparative analysis of business process modeling notations]. *Open Systems*, no. 8, pp. 28−32 (in Russian).
12. Gusejn-Zade S.M. (2003) *Razborchivaja nevesta* [The smart bride]. Moscow: MCCME (in Russian).
13. Mikheev A.G. Orlov M.V. (2011) Sistema upravlenija bisnes-protsessami i administrativnymi reglamentami [Business process and administrative regulation system]. *Programmnye Produkty i Sistemy*, no. 3, pp. 126−130 (in Russian).

Проектирование исполняемых бизнес-процессов как парадигма программирования

А.Г. Михеев

кандидат физико-математических наук,
доцент кафедры бизнес-информатики и систем управления производством
Национальный исследовательский технологический университет «МИСиС»
Адрес: 119991, г. Москва, Ленинский проспект, д. 4
E-mail: andrmikheev@gmail.com

В.Е. Пятецкий

доктор технических наук, профессор,
заведующий кафедрой бизнес-информатики и систем управления производством
Национальный исследовательский технологический университет «МИСиС»
Адрес: 119991, г. Москва, Ленинский проспект, д. 4
E-mail: 7621496@gmail.com

Аннотация

В статье рассматриваются приемы, применяемые при разработке бизнес-процессов, непосредственно исполняемых в компьютерной системе предприятия (исполняемых бизнес-процессов). Представлен опыт обучения элементам этой технологии, полученный в Национальном исследовательском технологическом университете МИСиС и Московском государственном университете экономики, статистики и информатики (МЭСИ) в течение двух лет преподавания процессных дисциплин в бакалавриате и магистратуре.

Одной из причин выбора процессного варианта автоматизации предприятия является уменьшение затрат на автоматизацию. При традиционной автоматизации сначала бизнес-аналитик описывает функциональность проектируемой системы в виде текста, а затем программист переводит это описание в программный код. Использование исполняемых бизнес-процессов позволяет в значительной степени избежать дублирования работы: в этом случае бизнес-аналитик совместно с заказчиком при помощи визуальных графических программных средств разрабатывает бизнес-процессы автоматизируемой функциональности, которые будут потом непосредственно исполняться в компьютерной среде. Схемы исполняемых бизнес-процессов представляют собой понимаемое человеком графическое описание соответствующей функциональности, при этом их не требуется переводить в программный код. Поэтому затраты на аналитическую деятельность в этом случае будут примерно такими же, а затраты на программирование — существенно ниже. При изменении условий бизнеса бизнес-аналитик может быстро изменить соответствующим образом схемы бизнес-процессов без участия программиста. Также во многих случаях бизнес-аналитик может самостоятельно (без участия программиста) разрабатывать новые бизнес-процессы. Поэтому стоимость разработки, сопровождения и поддержки такого ИТ-решения оказывается существенно меньше стоимости традиционного решения, а скорость разработки, внедрения, а также последующих изменений — существенно выше.

Эти преимущества (быстрее, дешевле, легче в поддержке и сопровождении) совпадают с преимуществами парадигмы объектно-ориентированного программирования по сравнению с парадигмой процедурного программирования. По аналогии мы можем назвать разработку программного решения на основе исполняемых бизнес-процессов новой парадигмой программирования относительно традиционного подхода.

Процессная автоматизация на основе исполняемых бизнес-процессов требует от специалистов — бизнес-аналитиков «процессного» мышления, отличающегося от мышления ИТ-специалистов, занимающихся традиционной автоматизацией предприятий. Кроме знания процессных нотаций, бизнес-аналитики должны уметь реализовать в виде исполняемых бизнес-процессов типичные ситуации, возникающие в бизнесе предприятия. В статье приведены методики, используемые при обучении студентов «процессному» мышлению.

Литература

1. Van der Aalst W., Van Hee K. Workflow management: Models, methods and systems. Cambridge, MA: MIT Press, 2004. 384 p.
2. Абдикеев Н.М., Данько Т.П., Ильдеменов С.В., Киселев А.Д. Реинжиниринг бизнес-процессов. М.: Эксмо, 2005. 592 с.
3. Тельнов Ю.Ф. Реинжиниринг бизнес-процессов: Компонентная методология. М.: Финансы и статистика, 2004. 320 с.
4. Калянов Г.Н. Моделирование, анализ, реорганизация и автоматизация бизнес-процессов. М.: Финансы и статистика, 2006. 240 с.
5. Вагнер Ю.Б. BPMS-эффект // Автоматизация в промышленности. 2009. № 7. С. 42−47.
6. Floyd R.W. The paradigms of programming // Communications of the ACM. 1979. No. 22 (8). P. 455−460.
7. Kuhn T.S. The structure of scientific revolutions. Chicago: University of Chicago Press, 1962. 172 p.
8. White S.A., Miers D. BPMN modeling and reference guide: Understanding and using BPMN. Lighthouse Point, FL: Future Strategies, 2008. 231 p.
9. Куликов Г.Г., Михеев А.Г. Особенности реализации процессного подхода и обучения управлению бизнес-процессами при помощи свободного ПО с открытым кодом // Открытое образование. 2011. № 4. С. 47−57.
10. Пятецкий В.Е, Михеев А.Г., Новичихин В.В. Система управления бизнес-процессами: основы разработки бизнес-процессов с помощью свободного программного обеспечения: Практикум. М.: МИСиС, 2013. 207 с.
11. Федоров И.Г. Сравнительный анализ нотаций моделирования бизнес-процессов // Открытые системы. 2011. № 8. С. 28−32.
12. Гусейн-Заде С.М. Разборчивая невеста. М.: МЦНМО, 2003. 24 с.
13. Михеев А.Г., Орлов М.В. Система управления бизнес-процессами и административными регламентами // Программные продукты и системы. 2011. № 3. С. 126−130.