REFERENCES

1. Buch, Grady. Object-oriented analysis and design with application examples on C+: Translated from English / G. Buch. – 2d issue – Moscow: Binom, 2000. – 560 p.: ill.
2. Kazimir V.V., Demshevskaya N.V. Formal object-oriented approach to complex system simulation // In collected articles: «Перша Міжнародна науково-практична конференція з програмування УкрПрог'98». – Kiev: Cybernetic centre of Ukraine NAS, 1998. – P. 593–598.
3. Tsapko S.G. E-network facilities of modeling and simulation of processes of functioning complex dynamic configurable systems in tasks of construction of computer simulators: Thesis of ... candidate of tech. science. – Tomsk, 2003. – 192 p.
4. Tsapko G.P., Tsapko S.G. Tarakanov D.V. Basic principles of E-network model construction of complex engineering system // Bulletin of the Tomsk Polytechnic University. – 2006. – V. 309. – № 4. – P. 152–157.
5. Tsapko S.G., Tsapko I.V. Object presentation of simulation models of complex engineering system subsystems in terms of E-network // Bulletin of the Tomsk Polytechnic University. – 2006. – V. 309. – № 5. – P. 167–170.

Received on 25.06.2007

UDC 519.876.2(004.4'22)

TECHNIQUE AUTOMATED OF DIAGRAM CONSTRUCTION IN BUSINESS PROCESS MANAGEMENT SYSTEMS

I.G. Ozerova, E.A. Dmitrieva, G.P. Tsapko, V.N. Vichugov

Tomsk Polytechnic University
E-mail: ira@aics.ru

Technique allowing reducing stages of analysis and design of application while implementing Business Process Management System (BPMS) has been suggested. It was possible due to elimination of enterprise activity examination stage and formation of business process models on the basis of structural functional models obtained as a result of reengineering project or developing quality management system. The required steps of model construction in BPMS were revealed. Appropriateness of business process modeling with the help of traditional meansand further use of models for transfer into BPMS by conversion was validated. Algorithm of automated transformation on the basis of processing XML-files of models was suggested.

In the radical reengineering becomes unacceptable method for improving business processes as its labor content does not allow reacting rapidly to changeable market demands. Corporate information systems of type *ERP* (*Enterprise Resource Planning*) introduced by reengineering results allow adapting to any enterprise structure but frequently long duration of reconfigurations disables enterprise to manuge their business process managing in real time.

Investigations of enterprise management principles from the position of process approach are widely given in scientific literature by both foreign and domestic authors [1–5]. At the beginning of the current century the process approach has got supporting in the form of software tools *BPMS* and now is called *BPM – Business Process Management* [6–9]. *Unify NXJ* (Unify), *Oracle BPEL Process Manager* (Oracle), *ActiveBPEL* (Active Endpoints) may be given as the examples of *BPM*-systems (and their developers). *BPMS* architecture including graphics editor, engine, monitoring module allows updating existing processes in the required rate.

A code in a special programming language, for example *BPEL* (*Business Process Execution Language*) corresponds to the diagram of the process in *BPMS*. The process in *BPEL* language itself does not fulfill any functions and intended exclusively for coordination (or *orchestration*) of web services. *BPEL* specification is approved as standard of OASIS (Organization for the Advancement of Structured Information Standards) [10]. Some part of developers uses this standard and others use their own nonstandardized languages of process description.

At present applications in *BPMS* are developed on the basis of user manuals which contain description of interfaces and installation steps. New concept requires novelty in its usage as well.

The suggested technique of process diagram construction in *BPMS* includes the following main stages:

1. Simulation of business processes using facilities of *CASE* (*Computer-Aided Software/System Engineering*).
2. Conversion of *CASE*-model into *BPM*-model.

Here and further *CASE*-facilities are implied as their subset which is intended for simulating business processes. For example, *AllFusion Process Modeler* (earlier *BPwin*), *ARIS Toolset. BPwin* supports the following methodologies: *IDEF*0 (*Integrated Computer Aided Manufacturing* (*ICAM*) *DEFinition language* 0), *DFD* (*Data Flow Diagram*), *IDEF*3; *ARIS Toolset – VACD* (*Value-added chain diagram*), *eEPC* (*extended Event-driven Process Chain*), *FAD* (*Function allocation diagram*), *IFD* (*Information flow diagram*) etc. Models implemented

according to these notations are also called structural functional models.

Use of such two-stage approach is substantiated in the following way:

- If enterprise has already examined its activity during process of reengineering or construction of quality management system and described business processes then business applications should be developed on the basis of existing models that supports reduction of development duration due to elimination of analysis and design stages. Models are transferred into *BPMS* by conversion techniques including automation facilities use. These techniques are developed on the basis of juxtaposition of traditional methodologies and *BPM*-specifications of business processes description. Juxtaposition by the example of *DFD* methodology and *BPEL* specification is examined in [11].
- If enterprise has not described before its processes or existing models lost their urgency then it is necessary to use notations of structural functional models for qualitative business processes analysis and design.

The last conclusion may be made as a result of comparative analysis of *CASE*-facilities and graphics editors of *BPMS* which includes the following main points:

1. Diagrams in *CASE*-facilities are more demonstrable and natural for visual perception as they are based on functional approach and notion of «blackbox» for which a specified set of output parameters corresponds to a certain set of input parameters. Actions in these diagrams are simulated by means of a single element – functional block in which the simulated function is described in some words. In *BPEL* a specified operation is fixed to each block and it is necessary to choose a type of block before imaging any activity on *BPEL* diagram. Reading *BPEL* diagram it is also necessary to know each block dedication. Besides, diagrams in *BPM*-systems include also blocks of exception (errors) processing required straight at performance that overloads diagram. Use of *CASE*-facilities at stage of analysis and design is easier for analyst.
2. In *BPM* systems there is no capacity to use phrases from several words in blocks, transitions names that forces to reduce them and results in diagram informativity lose. So, for example, transition name in model *Unify NXJ* can not be separated into several lines, therefore, it closes other objects of the model. Also in *BPMS* Russian is not always supported at development and performance of business processes. *CASE*-facilities have no such disadvantages.
3. On *BPM* diagrams it is not seen what kind of information is required and transferred for performance of one or another block inside business process and between services as well, as arrows point only to action sequence. At the same time arrows in *BPwin* point both to information flows transmitted between functional blocks and to sequence of function performance.
4. Enterprises aiming at increase of quality and efficiency of activity and having made a decision to construct quality management system and certify it turn to outer organizations which use just traditional facilities of business processes modelling for describing consumer processes. These faculities are intended specially for suchpurpose and won wide recognition (*BPwin*, *ARIS Toolset*). In this case experience in use of the given facilities increases naturally specialist efficiency at enterprise activity description.

Automation of *CASE*-model transformation into *BPM*-models is suggested to be fulfilled on the basis of parsing, analysis and conversion of *XML*-documents of processes. It became possible at appearance of model saving capacity in format *XML* in *AllFusion Process Modeler* (*BPwin*) version 4.1.4. Besides, *ARIS Toolset* gives an opportunity to export model *eEPC* into format *BPML* (*Business Process Modeling Language*) [12] based on *XML* and used for business process modeling. Languages of business process models in *BPMS* including *BPEL* are also based on syntax of *XML* language. Algorithm of automated transformation is shown in Fig. 1: firstly, *CASE*-model of business process is exported or saved in *XML* format; then *XML*-model is converted by automatic transformation block into *XML*-model in *BPM* language; after that *BPM*-model is updated and added with attributes if it is required and at the output the working business process which may be performed by *BPM*-system occurs. To implement automatic transformation block knowledge of *XML*-scheme of input model is required for its parsing for extracting transformed elements according to the obtained techniques of transformation also knowledge of *XML*-scheme of output model is requred to form it on the basis of transformation techniques.

To implement automatic transformation it is necessary to:

1) Obtain *XML*-schemes according to which *XML*-models are constructed. Not all the developers of the given products give such data; therefore, scheme generation is a separate stage.
2) Develop algorithms of processing elements of input XML-file.
3) Develop a system by programming tolls which obtains at the input the model of business process in the form of *XML*-document and at the output it gives *BPM*-model also in the form of *XML*-document.

Structural elements of output *XML*-document should be formed on the basis of its scheme and input document data in accordance with the developed business processes transformation techniques and processing algorithms.

Then work of automatic transformation block is examined by the example of *eEPC* → *Unify NXJ* transformation. As it was said before *ARIS* medium allows exporting model *eEPC* into file of *BPML* format. The simplified scheme of such file has a form as it is in Fig. 2 that means that the whole process of this file is concentrated in the main container `sequence` which may in its turn contain

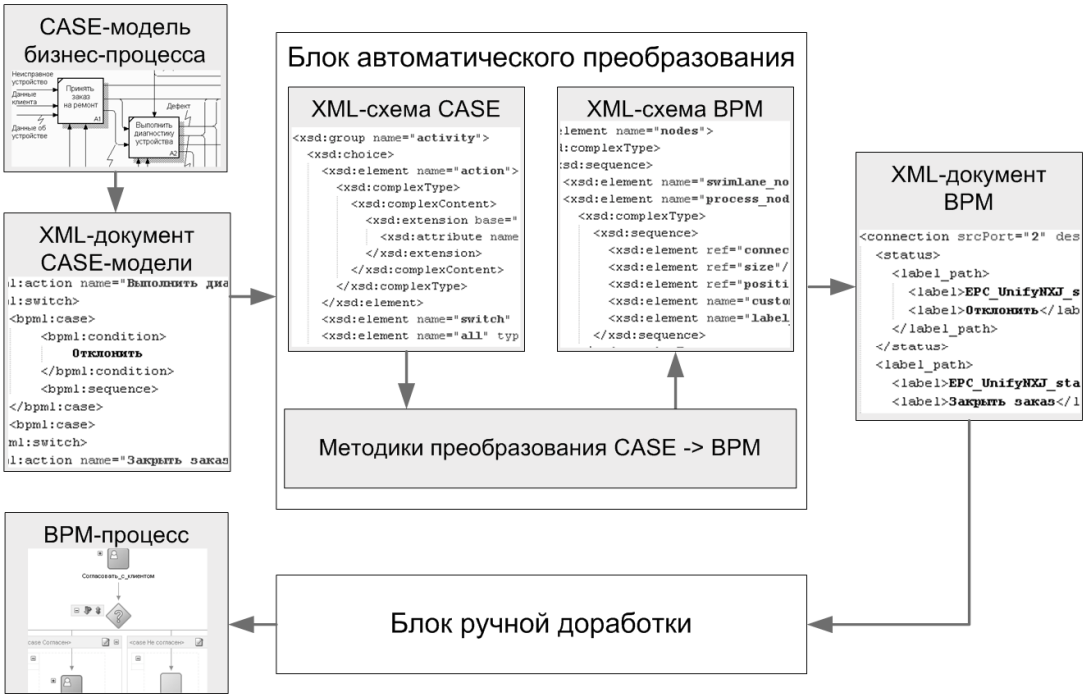

Fig. 1. *Algorithm of automatic transformation of CASE → BPM*
CASE-модели бизнес-процесса – CASE-model of business process; XML-схема CASE – XML-scheme of CASE; BPM-процесс – BPM-process; XML-документ CASE-модели – XML-document of CASE-model; XML-схема BPM – XML-scheme of BPM; Методики преобразования CASE – BPM – Transformation techniques CASE – BPM; XML-документ BPM – XML-document of BPM; Блок ручной доработки – Block of manual overpatching

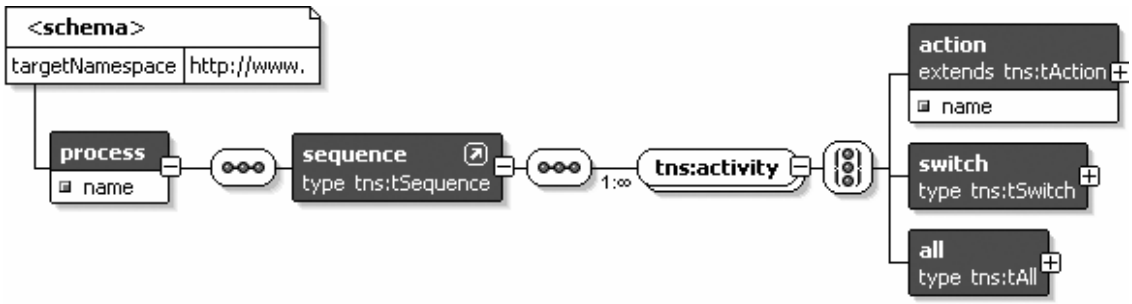

Fig. 2. *XML-scheme of BPML-file (block representation)*

elements of three types: `action, switch, all`. The specified elements correspond to the following elements of the model *eEPC*: function, Exclusive OR, AND.

Nodes `switch` and `all` are containers and have a structure showed in Fig. 3.

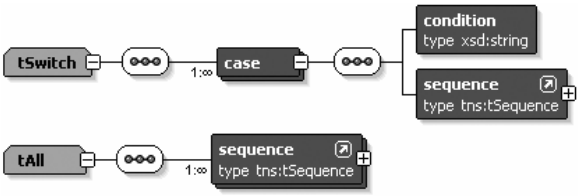

Fig. 3. *Block representation of BPML elements* `switch` *and* `all`

Each element of *BPML* should be singled out and treated in such a way that *XML*-file occurs at the output; it corresponds to *XML*-scheme of the model of business process in *Unify NXJ* the enlarged scheme of which looks like this (Fig. 4).

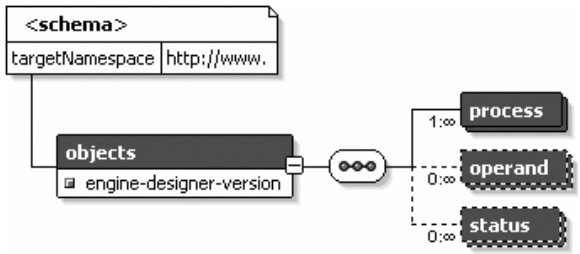

Fig 4. *XML-scheme of the model Unify NXJ (block representation)*

Algorithms of two first detailing level suggested for processing *BPML*-files are given in Fig. 5, 6.

Recursive approach to processing of *XML*-files of business processes allowed simplifying implementation of some functions. For example, to find the last process elements which should be connected with finite state it is not enough to find the last node inside node `sequ-`

ence as it can be node switch which can contain in its turn some branches sequence with its nodes. Then it can be checked out whether the current block is the last or not according to the following algorithm:

1. Find the last node in the main sequence.
2. Check out whether the current node action is the found last node.
3. If it is so then to create transition from the current node into the finite one.
4. If it is not so then to check out whether the last node is the switch node.
5. If it is so then to find the last node for each branch switch.
6. Check out whether the current node is the last node of the branch switch (recursive call).
7. See steps 3 and 4.

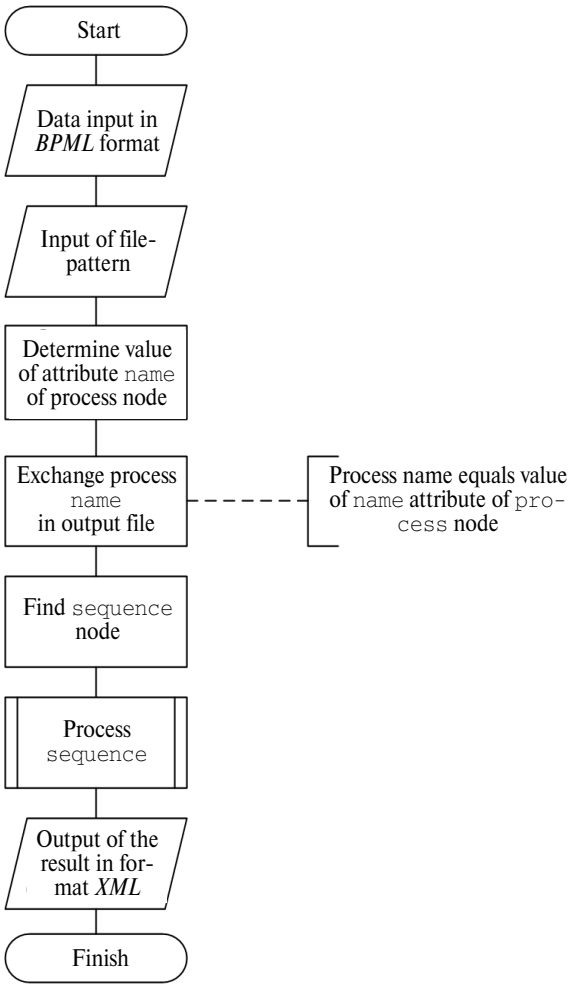

Fig. 5. *Algorithm of processing of XML-document in format BPML*

Software implementation of the block of automatic transformation of business process models is carried out in the *Borland Delphi* enviroment on the basis of the developed algorithms using components allowing analyzing *XML*-files. The result of software performance is *XML*-file which may be opened and executed in the system *Unify NXJ*.

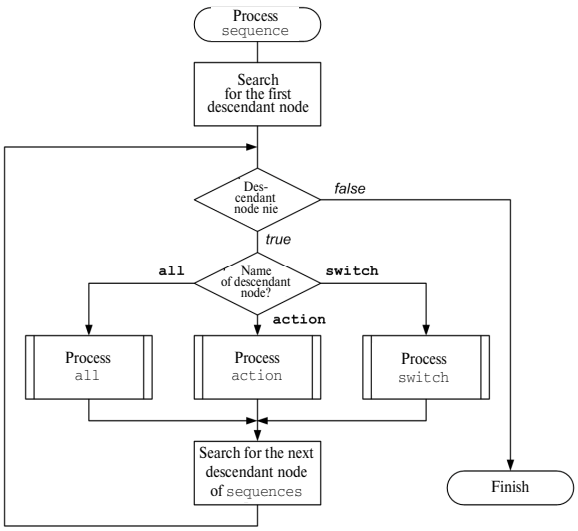

Fig. 6. *Algorithm of sequence element processing in BPML*

The main window of the developed software is shown in Fig. 7. In this window it is necessary to indicate name of the file containing business process model in format *XML* and intended for transformation and also the name of the file where the resulting BPM-model should be saved after clicking «Convert».

Fig. 7. *The main window of the program of process model transformation automation*

At the moment model transformation is implemented in one direction only (from structural functional models into BPM-models). Bidirectional transformation may be also useful for supporting models *BPwin*, *ARIS* in urgent state. Therefore, the development of additional module may be further task.

So, possessing powerful potential in operational changing of business processes, *BPM* systems do not allow importing models performed in *CASE*-facilities and do not possess such significant means of process description and analysis as *CASE*-facilities. Therefore, the latter should be used for qualitative analysis and design of business processes and then transfer models into *BPMS*. Hence the technique of process models development in *BPM*-systems including two stages was suggested.

The first stage consists in modeling of processes by *CASE*-facilities. For enterprises that have desckibel their activity this stage allows using cumulative experience in the field of their activity description and the obtained models. *The second stage* consists in conversion

of *CASE*-models into *BPM*-models. For this purpose the algorithm of automatic conversion of models in format *XML* into schemes of *BPM*-systems was suggested. The developed techniques of transformation of business process models and the developed algorithms of conversion of *XML*- documents of the processes into *BPMS* formats are in the basis of this algorithm. This algorithm allows performing a part of transformations in automatic mode by recursive processing of *XML*-processes. The given technique allows accelerating the deployment of process management on the basis of *BPMS*.

Conclusion

The technique allowing decreasing the stage of application analysis and design when deploying business processes management system was suggested. The order of constructing models in *BPMS* is given. Appropriateness of business processes modeling by traditional modeling tools with further use of the models for transfer into *BPMS* by means of conversion was justified. The algorithm of automated conversion on the basis of processing *XML*-files of the models was suggested.

REFERENCES

1. O'Liry D. ERP systems. Modern planning and management of enterprise resources. Choice, introduction, maintenance. – Moscow: Vershina, 2004. – 272 p.
2. Harington J., Esseling K.S., Nimvegen H.V. Optimization of business process: Documentation, analysis, management, optimization. – St. Petersburg: AZBUKA: BMicro, 2002. – 328 p.
3. Hammer M., Chumpy J., Corporation reengineering: Manifest of revolution in business. – St. Petersburg: St. Petersburg University Press, 1997. – 332 p.
4. Kalyanov G.N. Theory and practice of business process organization. – Moscow: SINTEG, 2000. – 212 p.
5. Repin V.V., Eliferov V.G. Process approach to management. Business process modelling. – 2d issue. – Moscow: RIA «Standarty i Kachestvo», 2005. – 408 p.
6. Svinarev S. Business Process Management: from idea to practical realization // PC Week / RE. – 2005. – № 34. – P. 39–41.
7. Belaichuk A. BPM test // System opening [Electron resource]. – 2006. – № 1. – Access mode: http://www.osp.ru/os/2006/01/380764, http://bpms.ru/library/articles/bpm-exam/index.html.
8. Smith H., Fingar P. Business Process Management: The Third Wave. – Tampa: Meghan-Kiffer Press, 2003. – 311 p.
9. Chang J.F. Business Process Management Systems: Strategy and Implementation. – [s.I.]: Auerbach, 2005. – 286 p.
10. Web Services Business Process Execution Language Version 2.0. OASIS Standard [Electron resource]. – 2007. – 264 p. – Access mode: http://docs.oasis-open.org/wsbpel/2.0.
11. Oserova I.G. Juxtaposition of traditional methodologies of describing business processes and language of their performance // Bulletin of the Tomsk Polytechnic University. – 2006. – V. 309. – № 7. – P. 205–208.
12. Corporation: languages of business process management. BPML [Electron resource]. – 2005. – № 41. – Access mode: http://www.iso.ru/journal/articles/390.html.

Received on 04.10.2007